WL-TR-96-3137

MIDAS: MULTIDISCIPLINARY INTERACTIVE DESIGN AND ANALYSIS SYSTEM - *ARCHITECTURE*

HARINI S. RAJAGOPALAN
XIAODONG LUO
RAMANA V. GRANDHI

Department of Mechanical and Materials Engineering
Wright State University
Dayton OH 45435-0001

DECEMBER 1995

INTERIM REPORT FOR SEPTEMBER 1994 -- DECEMBER 1995

Approved for public release; distribution unlimited

FLIGHT DYNAMICS DIRECTORATE
WRIGHT LABORATORY
AIR FORCE MATERIEL COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7552

NOTICE

THIS TECHNICAL REPORT HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION.

Victoria A. Tischler

VICTORIA A. TISCHLER
Aerospace Engineer
Design & Methods Development Section
Structures Division

Nelson D. Wolf

NELSON D. WOLF
Technical Manager
Design & Methods Development Section
Structures Division

Donald B Paul

DR. DONALD B. PAUL
Chief
Structures Division

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
DECEMBER 1995

3. REPORT TYPE AND DATES COVERED
INTERIM Sept 1994 - Dec 1995

4. TITLE AND SUBTITLE
MIDAS: Multidisciplinary Interactive Design and Analysis System -Architecture

5. FUNDING NUMBERS
CONTR: F33615-94-C-3211
PE: 62201F
PR: 2401
TA: TI
WU: 01

6. AUTHOR(S)
HARINI RAJAGOPALAN
XIAODONG LUO
RAMANA GRANDHI

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
DEPARTMENT OF MECHANICAL AND MATERIALS ENGINEERING
WRIGHT STATE UNIVERSITY
DAYTON OH 45435-0001

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
STRUCTURES DIVISION
FLIGHT DYNAMICS DIRECTORATE
WRIGHT LABORATORY
WRIGHT PATTERSON AIR FORCE BASE, OH 45433-7542
POC: V. A. Tischler, WL/FIBD, WPAFB OH; 937-255-56992

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
WL-TR-96-3137

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE. DISTRIBUTION IS UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This report presents the development of MIDAS, a graphical pre- and post-processor for ASTROS. The system is developed around I-DEAS using its Open Architecture. MIDAS displays structural responses like stresses, displacements under multiple boundary conditions and load cases. It also displays normal modes animation and concurrent aerodynamic and structural models. Present work concentrates on display of flutter modes, design optimization iteration histories and optimum variables distribution. "What-if" trade off studies is the future course of the work. MIDAS has been developed on Silicon Graphics workstation and is portable to other UNIX platforms.

14. SUBJECT TERMS
MULTIDISCIPLINARY ANALYSIS AND DESIGN, ASTROS, PRE- AND POST-PROCESSING, STRUCTURES AND AERODYNAMICS

15. NUMBER OF PAGES
206

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

LIST OF FIGURES

LIST OF FIGURES (continued)

LIST OF TABLES

FOREWORD

Preliminary design of aircraft structures is multidisciplinary, involving knowledge of structural mechanics, aerodynamics, aeroelasticity, structural dynamics and design concepts. These disciplines interact, effecting the performance of the structure. It is essential to consider the impact of several disciplines simultaneously to arrive at a satisfactory design. ASTROS (Automated STRuctural Optimization System) is a design synthesis tool for multidisciplinary analysis and design which involves extensive modeling procedures in identifying finite elements, boundary conditions and other design parameters. Airframe and automotive industries which plan to use ASTROS as their design tool can benefit a great deal if an automated parametric modeling technique is introduced into this software. A graphical design and analysis system was thus developed for ASTROS to address the issues of preliminary design of aircraft structures. The design and analysis system called MIDAS (Multidisciplinary Interactive Design and Analysis System) provides a graphics environment to carry out preliminary design studies and also provides a platform to conduct a 'what-if' parametric study on the structure until a satisfactory design is achieved. Graphic assisted finite element modeling techniques were introduced into MIDAS through integration with the commercial software I-DEAS (Integrated Design Engineering Analysis System) using its Open Architecture tools. The work presented here discusses the characteristics of MIDAS and describes in detail the linear statics, normal modes and unsteady aerodynamic analysis disciplines along with linear statics and normal modes optimization. Future work will document aerodynamic optimization discipline.

PREFACE

This technical report was prepared as part of the MIDAS development, outlining the programming structure and details of the graphic interface. Besides this report the **MIDAS User's Manual** is available as a separate document. The present version of MIDAS interfaces with I-DEAS Master Series version 2.1 and ASTROS version 12.0. MIDAS has been developed on Silicon Graphics workstations.

This work was performed by Wright State University. Dr. Ramana Grandhi is the principal investigator and the major contributors are Harini Rajagopalan and Xiaodong Luo.

Ms. Victoria A. Tischler is the Air Force Project Engineer and Dr. V. B. Venkayya was the initiator of the effort. Wright State University acknowledges the support of Mr. Raymond Kolonay, Dr. Robert Canfield, Mr. Greg Moster, Ms. Estelle Anselmo, Ms. Geetha Bharatram and Mr. Peter Flick (Technology Integration Area Core Leader).

Chapter One

Introduction

Aircraft design can be carried out in three phases, the conceptual design, preliminary design and detailed design. In the conceptual design phase, the basic configuration of the arrangement, size and weight, and performance are selected. It is an iterative process wherein new ideas are considered, a design is investigated and analyzed and checked for problems or checked to see if it meets specifications. If not, then the design is modified and the whole cycle is repeated.

Preliminary design begins after the conceptual stage. At this stage, the primary configuration is decided. During this phase, specialists in individual areas of design will design and analyze portions of the structure. Testing is done in areas such as aerodynamics, propulsion, structures, and stability and control. Preliminary design is the stage when the design configuration is frozen.

The detailed design stage is where the actual components to be fabricated are designed. For example, an aircraft wing is designed as a whole in the preliminary stage, whereas in the detailed design stage, ribs, spars and skins are designed independently for manufacturing.

The work presented here considers the preliminary design of aircraft structures. It discusses the characteristics and requirements of the preliminary design technique and presents an application that was developed for integrating and automating preliminary design.

Preliminary design of an aircraft structure involves the integration of several disciplines using sophisticated mathematical models in each field. Knowledge of structural

mechanics, aerodynamics, aeroelasticity, finite elements and design concepts are involved. Although theoretically, structures can be designed for individual disciplines, real world constraints often do not permit this approach. Several disciplines interact, effecting the performance of the structure. It is thus essential to consider the impact of several disciplines simultaneously to arrive at a satisfactory design.

It is evident that the preliminary design process can be time consuming and laborious. If the design takes multiple disciplines into consideration, it could be a complicated procedure. This complexity and time consuming nature of the problem poses a challenge to the designer to meet the fast growing needs of modern applications. Several steps have been taken by engineers to meet the challenges. Engineers have implemented iterative analysis and design processes as well defined algorithmic procedures. The possibility and feasibility of integrating and automating the algorithmic procedures have been studied and when found suitable, tools with integrated and automated design procedures have been developed. This process is constantly under development with enhancements to improve its characteristics and capabilities. Advanced techniques have been adopted during the development of these tools. Some of the most widely used techniques today in development of analysis or design synthesis tools include: Object oriented programming techniques to reduce the bulk of the code and provide a more efficient and time saving design methodology, expert systems to simplify the design process and assist the designer, interactive systems to give more control and monitoring facilities to the designer, graphic systems to enhance user-interaction and user-friendliness of the design process, etc. The following paragraphs will discuss some of the recent studies and developments in this field.

1.1 Literature Review

With the increase in the number of software applications, there is a lack of coor-

dination among software developers which results in the absence of standardization and difficulty in communication between one program to another. Transfer of information under these circumstances results in loss of manpower, time delays and potential for errors. Studies were made on the possibility of automating and integrating finite element based analysis and design procedures to alleviate this problem. Various enhancement techniques like knowledge based engineering, user-friendly modeling, visualization of information, etc, were considered along with automation and integration of software.

1.1.1 Applications for Integrating and/or Automating Design Procedures

Issues pertinent to finite element pre- and post-processing were studied by Benzley, et. al. [1] who made a study of present capabilities and suggested a few enhancements. The enhancements suggested by them include user-friendly geometric modeling, self adaptive element generation schemes, creation of finite element mesh with minimal input and visualization techniques for finite element post-processing. The work presented in this report has addressed these issues and has adopted these enhancements.

Tworzyllo and Oden [2] conducted a study of the feasibility of automation in computational mechanics. In particular they studied the methods of automation of finite element analysis which included algorithmic approaches and heuristic approaches based on knowledge engineering. They suggested that effort towards automation in computational mechanics was definitely feasible and promised great payoffs. The automation methodology suggested by them was the development of a computer environment in which algorithmic procedures were coupled with knowledge based systems. The authors felt that this kind of an approach could guide inexperi-

enced users through a maze of engineering software and computational procedures, assist experienced users in selecting mathematical models and computational strategies and control the quality of results and at the same time minimize efforts. Several integrated and/or automated tools were developed by engineers in the field of finite element analysis and optimization. Efforts were directed towards developing applications that provided a platform to conduct a design study, thus avoiding the problem of miscommunication or loss of information between different packages, erroneous transfer of data, loss of time and manpower, etc. Some of the integrated and/or automated systems reviewed are discussed here.

Tarrago, et. al. [3] developed CODYSIS, an integrated system for shape optimization which allows for the automated design procedure of linear elastic structures by integrating structural analysis, sensitivity evaluation and mathematical programming methods in a CAD environment. Emphasis is given on creating an appropriate geometric model with suitable design variables and a design variable linking procedure as well as an automated mesh generation and adaptive mesh refinement based on finite element error analysis.

Marlowe and Lindell [4] developed NASSTAR, a menu driven FORTRAN program that automates the model survey test/analysis correlation process by providing a translator between MSC/NASTRAN and STAR (test engineer's data processing code). It is intended to aid the structural analyst who must supply a test verified finite element model. NASSTAR pre-test support automates the translation of pre-test results from MSC/NASTRAN to STAR. NASSTAR's test procedure support section helps determine accelerometer locations using a pre-test finite element model. NASSTAR's post-test automates transferring test results from STAR to NASTRAN or PATRAN to view test mode shapes. Jiazhen [5] developed INTOAD, an interface between IRM (Integraph Rand-Micas) and ADINA. IRM is a finite element analysis

package of the Integraph CAD system. The IRM neutral file translator is a module which allows the user to create an ASCII file from an IRM model or read a properly formatted ASCII file to produce an IRM model. The ASCII file contains all the information defining the FEM model. Other finite element analysis programs such as ADINA, NASTRAN and ANSYS can use the information of the IRM neutral file. INTOAD has been developed to do this.

Rasdorf, et. al. [6] developed a computer aided analysis system in which a material database is integrated with several software including commercially available finite element analysis programs, pre-processors and tools for designing laminated composite materials. They developed a conceptual finite element material processing system for laminated fiber reinforced composites. The system focuses on assembling, manipulating and using composite materials resulting in the transfer of 2D and 3D composite material data into a finite element program. The system supports composites through testing, analysis and design. Arabshahi, et. al. [7] presented a system for CAD-FEA integration. The overall aim was to make the transition from solid model to idealized model for analysis an easier, robust and less time consuming task which otherwise is a time consuming task prone to errors and imperfections in modeling during the recreation of the FEA model from the CAD model. The authors proposed a system that would (i) have a robust and comprehensive product description system that would contain unambiguous multidimensional geometry data, environment conditions, design requirements, manufacturing costs, etc, (ii) have an intelligent semi-automated means for transforming the above information suitable for finite element mesh generation, (iii) have intelligent meshing routines to give feedback on meshing quality and suggest alternate strategies, (iv) have a series of finite element solvers to solve a wide range of problems and (v) have a post-processor to include the ability to associate results with the idealized model to allow for model modifications.

Stone, et. al. [8] developed an interactive pre-processor for structural design sensitivity analysis and optimization. The process of design parameterization and performance characterization are proposed as distinct design steps in the design process. These design parameterization steps enhance interaction between the design, analysis and manufacturing engineering groups that make up the current engineering design team. By using computer graphics and structural modeling software to define design parameters that characterize the design components, the pre-processor provides a natural way of defining acceptable classes of design. Along with the development of automated tools for design, several advanced techniques have been adopted to simplify and shorten the design process and make the implementation more efficient and viable. Some of the applications developed, which used techniques like object oriented programming, expert systems and graphic systems are discussed here.

1.1.2 Applications using Object Oriented Programming Techniques

Miki and Murotsu [9] developed an object-oriented structural analysis technique for modeling, analysis and conceptual design of advanced truss structures. In this approach, the entities in truss analysis were modeled as classes and their class hierarchy was established. The knowledge for a truss node, member, material and sectional shape were extracted and constructed as a knowledge base in an object oriented manner. Since the knowledge of truss structures and the related objects are already established, the designers do not need to take care of fundamental engineering knowledge and can devote themselves to creating new truss structures.

Caradona, et. al. [10] developed a finite element program that was written in C++ using object-oriented programming techniques in order to eliminate the risk of stagnating the program which has happened with many codes currently. They

provided the program with a powerful command interpreter that allows the user not only to introduce data but also to define the algorithms that will treat this data to obtain the result. Pidaparti and Hudli [11] developed an object-oriented analysis of finite element methods for dynamic analysis with a view of increasing the efficiency, robustness and state of the art function relation in the context of a CAD system for engineering design. This approach offers potential benefits of accessing the programs to tailor one's particular needs and the addition of software to the existing general purpose finite element programs.

Yu and Adeli [12] developed an object-oriented finite element modeling approach for the solution of complex engineering problems. For efficient processing of a myriad of data types generated in such analysis, an object-oriented enhanced entity relationship (EER) data model is developed. Class libraries created model the basic concepts and tools needed for the finite element analysis of engineering problems. The models class libraries developed were applied to the interlaminar stress analysis of composites. This type approach leads to highly modular and easily maintainable finite element software systems

1.1.3 Applications using Expert System Techniques

Some computer codes associated with the design procedure are complex, requiring extensive domain proficiency and knowledge on the part of the user, thus limiting the tools to experienced and knowledgeable users only. Hence, system engineers started to explore new techniques to implement the design process. They tapped the potentials of the Artificial Intelligence field in solving this problem and came up with Knowledge Based Expert Systems. By applying Knowledge Based Expert Systems to finite element based design problems, knowledge could be communicated to the

user to aid in conducting design studies in an easy and quick manner. Some expert systems developed for finite element analysis and design are discussed here.

Chandu, et. al. [13] developed an expert system for the Vulnerability Analysis of Aerospace Structures Exposed to Lasers (VAASEL). Issues pertaining to finite element based failure analysis of laser exposed structures were addressed. The system was designed predominantly with a graphic mode of knowledge representation. The system was designed to be user-friendly with respect to knowledge communication and was intended for a novice analyst.

Labrie, et. al. [14] developed an expert system to monitor a full time finite element simulation. The expert system acts as a consultant during the simulation, advising the user in mechanical analysis, the prescription of boundary conditions, assessment of numerical results and choice of various numerical parameters available in the simulation. It performs well in detecting errors and inconsistencies during the simulation preparation and analyzes the reliability of the result.

The expert systems discussed above act as consultants. Expert systems have also been developed to actually develop models of the structure for design and analysis and also validate the models. Kang and Haghighi [15] developed an intelligent finite element mesh generator called INTELMESH for 2D linear elasticity problems. They incorporated the information about the object geometry as well as the boundary and loading conditions to generate finite element mesh which is more refined around the critical regions of the problem domain. INTELMESH is fully automatic and allows the user to define the problem domain with a minimum of input. Wang, et. al. [16] developed an intelligent systems environment called IDIDE (Integrated Distributed Intelligent Design Environment) for automated conceptual design. This system has five stages. Stage one was to define the problem in which depending upon the application environment and purpose of the design specified, the system picks

up an appropriate function (or task) from a knowledge base. Stage two is when the structure to execute the function (or task) is selected from another knowledge base. A detailed design of this structure will be developed in the further stages to execute the particular task. Stage three is the parametric design stage where a detailed description of the structure can be completed by using a design model in model the knowledge base. Stage four is the analysis stage using numerical methods, and stage five is for comprehensive evaluation.

Chey and Zeng [17] developed a strategy to automate a finite element modeling system. Most of the expert systems developed aim to provide the user with expert consultation but rarely are a part of the pre-processing a system. The automated finite element modeling system begins with the geometry and attribute definition of structures, applies the object representation model and generates the model. Alternately, it makes use of a mesh generation system and extends it further to automate the modeling process by considering the problem attributes. Zhang, et. al. [18] developed software to automate the generation of quadrilateral mapping elements for application in shape optimization. They presented a method to define the structure's boundary of the plane continuum that is modified dynamically during the process of shape optimization. Heuristic rules and techniques from artificial intelligence were applied to generate the geometric model described by the design element methods, i.e. the structure was subdivided into large quadrilateral mapping elements which were necessary for mesh generation. This was an important step towards a fully automated computer aided shape optimization system.

Grandhi, et. al. [19] developed ETOP, an expert system for trajectory optimization. ETOP uses aircraft performance analysis expertise and experience as heuristics to aid users in generating input data models required for different hypersonic vehicles flying at optimum trajectories.

Daniel [20] developed an expert system for validation of displacement and rotation boundary conditions on finite element models. Time is wasted in finite element analysis because of errors in the data that can be identified using expert system techniques of programming. A data validation program was hence developed for detecting common errors in boundary conditions that create unwanted rigid body motion or errors that incorrectly represent the symmetry. An expert system shell is used to develop rules to test the finite element data. The rules are then translated into a more efficient procedural form.

1.1.4 Applications using Computer Graphics

An integral consideration for an analysis or design synthesis tool is the human computer interaction. The knowledge communication between the system and the user has to be effective, which means a good user-interface is required. Swanson [21] states that since its inception, finite element analysis has expanded in terms of ease of use, integration with other Mechanical Computer Aided Engineering (MCAE) technologies and advanced analysis capabilities. Advances in user-interfaces, graphics and solid modeling make finite element analysis more user-friendly and more appropriate for wider use. Several applications that have been developed in a graphics based environment for conducting finite element analysis or optimization are discussed here. These applications have been developed for a wide range of components ranging from space structures to pressure vessels, sheet metals, turbine blades, beams, frames, etc. The graphic user interfaces have been developed on various platforms like UNIX X Window environment, Macintosh, and PC's.

George, et. al. [22] developed the Multidisciplinary Integrated Design Assistant for Spacecraft (MIDAS) specifically for spacecraft design. The system performs the

following tasks: (i) allows the user to enter the design methodology for designing their element of the project and interconnect the elements in the most natural graphic way, (ii) uses as input the requirements levied on each element by the project and by the elements being designed by the other design engineers, (iii) uses commercial tools such as NASTRAN and SPICE, resident on different computer platforms, as part of the certification and eventual execution of the methodology, (iv) in addition, uses user-defined FORTRAN and C codes and (v) allow algorithms of multidisciplinary optimization of the system to be integrated by allowing the operator to compare different techniques for searching the solution space.

Chandu, et. al. [23] developed RELOPT (RELiability based structural OPTimization), a Motif graphic user interfaced application in an X window environment for comprehensive reliability based optimization. RELOPT is interfaced with an efficient safety index calculation code and a reliability optimization code to determine the probability of failure of the structure. The graphic user interface in RELOPT enhances the visual perceiving capabilities and analysis/design input and output data interpretation facility and saves considerable effort and time on the part of the user at the same time minimizing the possibility of modeling errors.

Zheng, et. al. [24] developed FEView, an interactive visualization tool for finite elements based upon an object-oriented graphics library. The visualization tool acts as an external module to an interactive program, Geomview, for viewing and manipulating geometric objects. The graphic user interface has been built on top of the Forms Library, a graphic user interface toolkit for SGI workstations. Geomview has been used as a display engine to which the code FEView acts as an external module.

Fox [25] developed a Probabilistic Design System (PDS) for gas turbine blades. There were four major criteria for developing this system. One was to base the design system on existing design tools used in deterministic optimization. That was because

use of a 'new' or unfamiliar or not widely accepted design practice could lead to the demise of the Probabilistic Design System. The second criterion was that the design system could take a small amount of additional time over the time required for a deterministic design. Next, the results of PDS must be quantifiably accurate. Finally the PDS was required to be user-friendly in order to avoid its stagnation due to interaction with the file syntax and the main program. PDS was required to have the feel of any deterministic optimization program and in fact to make the probabilistic design information as transparent as possible. Hence PDS was developed with a Motif graphic user-interface which makes the entire design process very easy.

Finite element analysis may produce large amounts of response outputs that can obscure an understanding of model behavior simply by looking at the numbers produced by numerical simulation. Karupurapu and Bathurst [26] developed a finite element post-processing program to analyze response data produced by a finite element program. The post-processing program could display selected aspects of model response in a graphical form which could enhance the interpretation of results produced by the finite element analysis.

Kumar, et. al. [27] developed an interactive graphic software for analysis of rotationally symmetric structures like cooling towers, chimneys, etc using the program ROTSYM. The idea was to provide a graphic medium for data preparation and comprehension of output to save time. It helps make visual checks on structure modeling, deformation and stress contours. Parkinson, et. al. [28] developed OPTDESX, an X window based optimal design software system. The software is designed to facilitate real time interactive optimization of engineering problems. Using point and click operations, a designer can rapidly define and explore an optimization problem. The software supports continuous, discrete and mixed optimization, sophisticated problem setup, multiple objectives and integrated graphics.

Lucas and Davis [29] developed MacPASCO, an interactive graphic pre-processor for structural analysis and sizing. The pre-processor creates input for PASCO, an existing computer code for structural analysis and optimization. By using a graphic user-interface, MacPASCO simplifies the specifications of geometry and reduces input errors, thus making modeling and analysis more efficient. The user draws the initial structure on the computer screen and uses a combination of graphic and text input to refine geometry and specify other aspects. A graphic user interface acts as a visual aid, eliminating the tedious text-based input and errors.

Santose, et. al. [30] developed an interactive post-processor for structural design sensitivity analysis and optimization sensitivity display and for a what-if study. A menu-driven multiwindow design workstation was developed. The design workstation supported (i) display of design sensitivity information, (ii) what-if studies, (iii) determining trade-offs and (iv) performing interactive design optimization. Graphic capabilities of established finite element pre- and post-processors were used in providing visualization of results.

1.2 Scope of the Work

As can be seen from the literature review, several approaches have been taken to simplify the design task. Efforts have been directed towards making analysis and design procedures as useful, user-friendly and as less laborious as possible. But from the works reviewed so far, one factor that becomes evident is that most applications are not comprehensive in nature. They address either analysis or design, pre-processing or post-processing, with very few of them having integrated all these aspects. Although some applications have integrated pre- and post-processing, they have not addressed the issues of modeling efficiency in terms of ease and accuracy or user-

friendliness. Also, several of them concentrate on a particular class of structures, thus losing generality.

Taking a step towards addressing all the above mentioned issues comprehensively, (integration of analysis and design pre- and post processing, improving modeling efficiency, increase user-friendliness and providing an environment to address a wide range of structures), a design and analysis system was conceived to address the issues of preliminary design of aircraft structures. The design and analysis system called MIDAS (Multidisciplinary Integrated Design and Analysis System) provides an environment to evaluate the structure until a satisfactory design is obtained and conducts a preliminary design procedure, iteratively if necessary. The major components that contribute to an entire design cycle in MIDAS are: Initial configuration, product simulation, interaction with analysis and design solver, design evaluation and redesign (if necessary). An approach has been taken in developing this system, in which user-interaction for the design study is highly graphics-oriented. The conventional dialog interface is minimized. The benefits of this approach are many. A graphic medium of communication enables the user to visualize concepts and design changes better than a conventional dialog communication. Also, it is very easy to spot modeling errors. The other important advantage of this approach is the ease of modeling. Large structures require enormous time if conventional modeling techniques are used. A graphic modeling technique reduces the engineer's preparation time significantly.

The purpose of MIDAS is to make multidisciplinary preliminary design and analysis simple and efficient using a graphic platform. It is an interactive graphics based design environment for automating the design problem. MIDAS allows interactive modification of free parameters in the initial design stage to study the impact on the structural response. If the response is not satisfactory, it allows parametric variation and redesign. The design package ASTROS (Automated STRuctural Optimization

System) developed at Wright Patterson Air Force Base is used by MIDAS to conduct the design study and the entire process is carried out in an iterative manner, with design changes recorded in each iteration. Since MIDAS is targeted to be a graphic environment, it is developed on a graphic platform. The powerful mechanical design automation graphic tool I-DEAS (Integrated Design Engineering Analysis System) developed by Structural Dynamics Research Corporation (SDRC), has been selected as the platform for developing MIDAS. Using Open Architecture software provided by SDRC, MIDAS has been totally integrated with I-DEAS to provide a graphic design environment. In addition to I-DEAS, a Motif user-interface has been used by MIDAS to communicate with the user. OSF/Motif is a user-interface standard and serves as a style guide for the look and feel and programming interface for the Motif toolkit. Motif is now accepted as an industry standard [31].

This report will discuss the initial configuration, product simulation and interaction with the solver in detail for the statics, modes and aerodynamics disciplines. The design evaluation aspects will be presented for the linear statics and normal modes disciplines.

1.3 Motivation for MIDAS Development

There are multitudes of analysis and design tools commercially available today, and a user is faced with the choice of selecting the most appropriate software to suit ones needs. The factor that comes next to accuracy that can increase a software's viability is its ease of use. The user will obviously be inclined to choose a package that is easy to use and requires minimal effort and time in terms of its modeling techniques.

ASTROS is a software tool developed for structural analysis and design and is

being planned for future commercialization. It is a multidisciplinary comprehensive software with a vast domain knowledge ranging from structural analysis to aerodynamics and optimization. Modeling using ASTROS requires extensive time consuming procedures. The time factor is more pronounced when the structures are very large in size, and the user may spend days in building a model which is free of errors. Recent research efforts in ASTROS have been directed towards enhancing its characteristics and capabilities along with easing the modeling technique. Yurkovich [32] conducted a study to determine the applicability of Taguchi technique coupled with the ASTROS code for use in optimum wing design. It was found that the procedure could be used effectively to determine the optimum wing internal and external geometry for minimum weight. The Taguchi technique was used to select the required computer runs to be made. An interactive code ASTRIGS [33], a software for partial automation of aircraft structural analysis and optimization data generation, was used to develop the input required for ASTROS code. The ASTROS code was then used to determine the minimum weight, and Taguchi technique was used to determine the optimum configuration. Luo and Grandhi [34] developed an application that introduced design capabilities in ASTROS that considered reliability as a design constraint. The uncertainties and randomness coming from the finite element model, material properties, boundary conditions, etc, were modeled and a constraint was imposed on the structure to meet a required level of reliability along with other displacement, stress and frequency constraints. The methodology developed improved the reliability of optimized structures. Sarma, et. al. [35] developed a knowledge based advisor for ASTROS. The multidisciplinary knowledge involved in structural analysis, aerodynamic analysis and numerical optimization were considered. The system generates analysis and design models for steady and unsteady aerodynamic disciplines along with statics and modal analysis. This system is presently lacking in any kind of

graphic mode of the design evaluation.

Aerospace and automotive industries which plan to use ASTROS as their design tool can benefit a great deal with a software like MIDAS. MIDAS can aid an ASTROS user in generating the model in a fraction of the time in comparison to conventional model generation methods. The graphic mode of model generation helps address the issues of speeding up the modeling procedure and making it simpler than the traditional modeling method for ASTROS. MIDAS supplements ASTROS capabilities by providing parametric modeling, analysis and design capabilities.

The other important feature considered in MIDAS is a graphic user-interface. The role of user-interface techniques in increasing the feasibility and usability of software cannot be undermined. Lack of an effective user-interface could diminish the interest of the user in the package and make it non-operational. At the same time, an efficient user-interface could motivate the usage of a package and can speed up the user-interaction phase, thus saving a lot of time and effort. With the advent of many user-friendly software, graphic user-interfaces have become part of most applications and users are beginning to expect applications to have polished user-interfaces that are easy to use, so that the user spends less time on familiarizing with program syntax and semantics. Based on this existing demand, MIDAS has been made highly graphics oriented in user-interface aspects. It has been developed on I-DEAS platform which offers a very easy-to-use graphic interface.

The following chapter will give a brief overview of ASTROS (solver) and I-DEAS (modeler) along with a brief discussion on their relation with MIDAS. The subsequent chapters will discuss issues pertinent with the development of MIDAS and its characteristics along with some examples.

Chapter Two

Overview of the Solver and the Modeler in MIDAS

2.1 Solver: ASTROS (Automated STRuctural Optimization System)

ASTROS (Automated Structural Optimization System) is a finite element based system that was developed at Wright Patterson Air Force Base to provide automated analysis and design software. It was developed for the preliminary design of structures. Initial configuration and materials for the structure are selected and the task in ASTROS is to determine the structural sizes that will provide an optimized design while satisfying numerous requirements [35, 36, 37, 38].

ASTROS is a multidisciplinary analysis and design system. Key disciplines influencing the design of structures, have been implemented in this code. Various disciplines effect the design of structures and in the real world they act together to influence the performance. It is therefore convenient to have a package that can deal with multiple disciplines simultaneously. ASTROS has this capability. Addressing multiple disciplines has been made possible by a high level language MAPOL (Matrix Analysis Problem Oriented Language) which enables interaction among the various modules created for the different disciplines. Data is transferred between different modules using CADDB (Computer Aided Design Data Base) which was developed for ASTROS. ASTROS is a comprehensive package which has both analysis and optimization capabilities. Multiple boundary conditions and within each boundary condition, multiple load sets can be handled by ASTROS for large scale problems

and a wide range of problems.

The major disciplines in ASTROS include Statics, Modal Analysis and Aerodynamic analysis and optimization. In the design task, the objective is to minimize weight subject to strength, displacement, flutter or frequency constraints. The design variables are the geometry of the structure which include thickness and cross-sectional areas.

The user directs ASTROS through an input data stream composed of commands to attach run time database files and multiple data packets. Each packet of data contains information required to execute ASTROS. ASTROS' input stream has three main data packets; namely, the executive control, solution control and bulk data. The hierarchy of appearance of the above mentioned input blocks is fixed. Executive control appears first, followed by solution control and bulk data. The basic architecture of ASTROS' input stream showing the hierarchy of input information is shown in Figure 1.

The executive control packet begins with an ASSIGN DATABASE entry, supplying the name of the database file and its parameters to ASTROS. The database files are used during ASTROS run time. The database contains input and output information which can be used later, if necessary. The database name with its password (to open the database), status (NEW/OLD/TEMP) and installation dependent optional parameters are included in this packet as shown in Figure 1.

The solution control begins with the command SOLUTION. It contains commands to select data to be used for all the cases in the analysis/design. It selects the specified data set from the bulk data packet. ASTROS is not only a design tool, but also can be used just for analysis. Thus there are two subpackets, OPTIMIZE and ANALYZE as shown in Figure 1, which can be used either together or separately. The OPTIMIZE subpacket can contain multiple boundary conditions and multiple disci-

plines (indicated in Figure 1 by Boundary 1, Boundary 2, Discipline 1 and Discipline 2). This subpacket contains reference to the discipline, constraints to be applied to the problem along with the physical boundary conditions. The ANALYZE subpacket specifies the type of analysis to be performed on the structure along with the physical boundary conditions. Multiple boundary conditions and multiple disciplines are once again enabled for analysis as shown in Figure 1. Output requests are made in the ANALYZE and OPTIMIZE subpackets.

The bulk data packet contains the finite element model, the aerodynamic model and the design model along with engineering data for conducting the analysis/design as indicated in Figure 1. It contains information on the finite element model, boundary conditions, aerodynamic elements and other parameters such as mach number, dynamic pressure, material and physical properties, applied loads and design data like constraint limits, initial value of the design variables, etc.

2.1.1 ASTROS Modules in MIDAS

MIDAS, which is as a graphic interface for ASTROS, will consider all the key modules present in ASTROS; namely static, modal and aero analysis and optimization. All the three input data blocks, namely, the executive control, solution control and bulk data will be generated by MIDAS adhering to the hierarchial levels of input block appearance. ASTROS modules selected for MIDAS are based upon user requirements. The static, modal and aero modules are the most frequently used modules in ASTROS for both analysis and optimization. The static and modal analysis modules were developed first in order to lay the basic foundation and infrastructure for subsequent modules. The other modules require most of the information from the static and modal analysis modules, in addition to boundary conditions and other data specific to them.

2.2 Modeler: I-DEAS (Integrated Design Engineering Analysis Software)

I-DEAS (Integrated Design Engineering Analysis Software) is a package developed for conducting mechanical design and analysis [39, 40, 41]. It is a package with a graphic user-interface and provides graphic means of generating the model. It is comprised of several applications, and within each application, a number of tasks. The applications in I-DEAS include: Design, Drafting, Simulation (Finite Element), Test, Manufacturing, Management and Geometry Translators. The I-DEAS application for our interest is "Simulation". MIDAS is integrated with the "Simulation" application, since MIDAS uses the I-DEAS "Simulation" as a means of generating the finite element data for ASTROS. The use of the "Simulation" application for MIDAS will be described in the forthcoming chapters with the tools used for MIDAS development.

The "Simulation" application of I-DEAS performs product simulation using finite element methods. The structure is divided into a mesh of elements, and structure stiffness, displacements, stresses and natural frequencies are computed for a given loading, material property and boundary conditions. The tasks affiliated with "Simulation" include Master Modeling, Master Surfacing, Meshing, Beam Sections, Boundary Conditions, Model Solution, Optimization and Post-processing as shown in Figure 2. These are the tasks used by MIDAS, although there are several other tasks under "Simulation". MIDAS uses only the pre- and post-processing capabilities of I-DEAS and uses ASTROS as an external solver. The element generation, material and physical property definition, loading and boundary conditions features of I-DEAS form the pre-processing phase in MIDAS. The model is solved for displacements, stresses, frequencies and other quantities by the external solver ASTROS. The post-processing phase is once again carried out in I-DEAS for MIDAS.

Part geometries can be created using the Master Modeler and/or Master Surfacing tasks in "Simulation" as shown in Figure 2, which forms the design phase. It is to be noted here that the user can straight-away generate a finite element model without having to go through the design phase. For some models, which are small in size, this method might prove to be effective. It is however recommended that for models which are large in size and have intricate geometry, the user starts with the design first. One other advantage of using the design phase first would be that it is easy to modify the model if necessary. MIDAS enables parametric studies wherein response of the structure to geometric variation can be studied. This requires many modifications to be made to the structure, which can be easily done in the design phase.

Initial design concepts are developed in the Master Modeler task. Parts can be created in two ways. One way is to use wireframe geometries like points, lines, arcs, circles and splines which can be extruded or revolved to get the desired shape. The other way is to use primitives from the standard part catalogue. Primitives are automatically created by specifying their characteristic parameters. They include blocks, cones, cylinders and spheres. These features of the Master Modeler can be used in various combinations along with construction operations like cut, join and intersect to design various structural components. Master Surfacing is an advanced modeling technique provided by I-DEAS to create complex curves and surfaces with overall shape control.

The design model created can be transformed into a finite element model by transferring the part geometry to the Meshing task as shown in Figure 2 or the finite element model can be created directly in the Meshing task. The meshing task includes node and element creation and material and physical property definition for the structure. Both manual and automatic node and element generation are possible. Nodes can be generated manually by either keying in or copying or reflecting a set

of nodes. These methods of manual node generation are possible whether the model has been generated in the Master Modeler or in the Meshing task itself. Elements can be generated manually by picking the nodes that connect to form the element or by mesh generation. In the latter case, the type of element, mesh size or number of elements on each side of the surface have to be defined before meshing to generate nodes and elements. Both free and mapped meshing options are available.

The cross-section type and dimensions are defined for rod and beam elements in the Beam Section task. The finite element model is then brought into the Boundary Condition task as shown in Figure 2. Loads and boundary conditions are applied in this task. I-DEAS allows restraint, constraint, degree of freedom and load sets to be created. Restraints are used to restrain the model to the ground, whereas constraints constrain the model to the other nodes, not to the ground. They are used to impose symmetric boundary conditions or special relationships between nodes. Kinematic degrees of freedom are defined for dynamics problem to reduce overall kinematic freedoms where no boundary conditions exist in a particular direction. Mechanical forces and moments, gravity and thermal loads can be defined in I-DEAS. Individual sets are created for restraints, constraints, degrees of freedom and loads. I-DEAS allows the creation of any number of these sets, although it is not necessary that all of them be applied to the problem. The user has the option to chose the sets in any combination applicable to the problem and create sets (I-DEAS boundary sets). Thus a boundary set can contain either restraint, constraint (in combination with at least a restraint or degree of freedom set), degree of freedom and load sets or all of them. Here again, all the boundary sets created need not be applied in the problem. One or more than one set can be applied, leaving out the ones not wanted. The selection of these boundary sets is done in the Model Solution task of I-DEAS. The boundary sets are passed on to the Model Solution task as shown in Figure 2. Each boundary set

is placed in a Solution Set in the Model Solution. Although analyses sets are created in Model Solution, this task is not used to solve the problem. Instead ASTROS is used as an external solver. MIDAS reads in the analysis set information from Model Solution as shown in Figure 2. Although the boundary condition information can be accessed from the Boundary Condition task itself, Model Solution was selected in order to provide the user with the advantage I-DEAS offers to create as many sets as wanted and to apply one or a few sets at a time.

The optimization task in I-DEAS can be either geometry based or finite element based. In MIDAS, optimization will be finite element based. The optimization goal is to minimize the weight of the structure subject to certain constraints. The constraint can be on stress, displacement or frequency. The physical variables such as thickness and cross-section area are the design variables. The Optimization task in I-DEAS creates design sets using the boundary sets shown in Figure 2 and passes on these sets to MIDAS. Like the analysis task, MIDAS will not use the I-DEAS solver, but instead uses ASTROS.

The post-processing task of I-DEAS will act as a post-processor for MIDAS too. The post-processing task receives results obtained from the analysis/design using ASTROS and displays the results as shown in Figure 2. I-DEAS also lets results be brought in from external solvers. The MIDAS post-processor is programmed to get results from the ASTROS database after the execution of analysis or design. Results such as displacements, stresses, strain energy, eigenvalues and forces are available.

2.2.1 I-DEAS Adaptability to ASTROS

Prior to developing MIDAS, it was essential to ensure compatibility between the two software by drawing comparisons. Compatibility here meant common analyses and design modules between the two software. If this compatibility was established,

it could lead to a direct translation process. In absence of this, other means of translation had to be developed.

A list of engineering modules present in I-DEAS, ASTROS and MIDAS are shown in Table 1. As mentioned in previous sections, MIDAS includes statics, normal modes and aero analysis and design. I-DEAS provides modules for statics, normal and constraint mode dynamics, linear buckling, heat transfer and potential flow. Statics and normal mode dynamic analysis are modules common to both I-DEAS and ASTROS as shown in Table 1. I-DEAS has an optimization module common with ASTROS and provides the minimum weight of structures for strength, displacement and frequency constraints. ASTROS, at present, does not solve heat transfer and potential flow problems. I-DEAS does not have an aero analysis or a design module.

As can be seen from the previous observations, static and normal modes analysis and design are common to both software. Translating information from these modules could be direct. The area where incompatibility was found was the analysis and design for the aerodynamic module of ASTROS. I-DEAS does not have aero analysis or design capabilities, whereas aero analysis and design are key modules in MIDAS.

It is to be noted here that although I-DEAS allows an interface with external finite element packages and solvers, it will not allow the programmer to build any additional features into it. Moreover, in the case of MIDAS, the problem is not to build an external aero analysis/design solver, since ASTROS is being used for this purpose. Hence the problem faced was to build an aero model in I-DEAS. An aero model requires defining aerodynamic elements, airfoil properties and aerodynamic parameters.

Some ways of dealing with this incompatibility were considered. One of them includes having an user-interface wherein the user in an interactive session will supply the information on aero analysis, and MIDAS will generate the aero model based on

this information. Efforts have been made to minimize this user-interaction. Internal computations have been done by MIDAS wherever necessary which will greatly minimize the user input. The other technique considered was mapping information. Mapping refers to matching information unknown to I-DEAS with quantities known to it. In other words, I-DEAS is made to 'believe' that an unknown quantity is a known quantity. For example, an aerodynamic model was superimposed on a structural model by displaying two models simultaneously. I-DEAS 'believes' both models to be structural models, whereas MIDAS interprets one of them as structural and the other as the aerodynamic model.

Even among compatible modules, there were some differences between the two software in the way elements were defined or design sets were created. In analysis sets, the major discrepancy was that I-DEAS did not allow multiple disciplines and multiple boundary conditions whereas ASTROS did. To deal with this incompatibility, the following strategy was developed: I-DEAS allows the user to create any number of analysis sets as desired, although it is not necessary that all of them be applied. Taking advantage of this fact, the user can create analyses packets and each of these could have different boundary condition sets and could belong to a different discipline. Now MIDAS will interpret all these sets as belonging to the same analysis set in ASTROS, thus the information that goes to ASTROS will have multiple boundary conditions and multiple disciplines. Apart from this, some information was missing in I-DEAS that was needed by ASTROS for which user-interface concepts were used. The details of the MIDAS architecture are discussed in later chapters.

Engineering Modules	I–DEAS		ASTROS		MIDAS	
	Anal	Opt	Anal	Opt	Anal	Opt
Linear Statics						
Normal Modes						
Constraint Modes						
Aerodynamics						
Buckling						
Heat Transfer						
Potential Flow						

Modules Present

Modules not Present

Modules common to ASTROS, I–DEAS and MIDAS

Modules common to ASTROS and MIDAS

Table 1. List of Engineering Modules and Common Modules Between I–DEAS, ASTROS and MIDAS

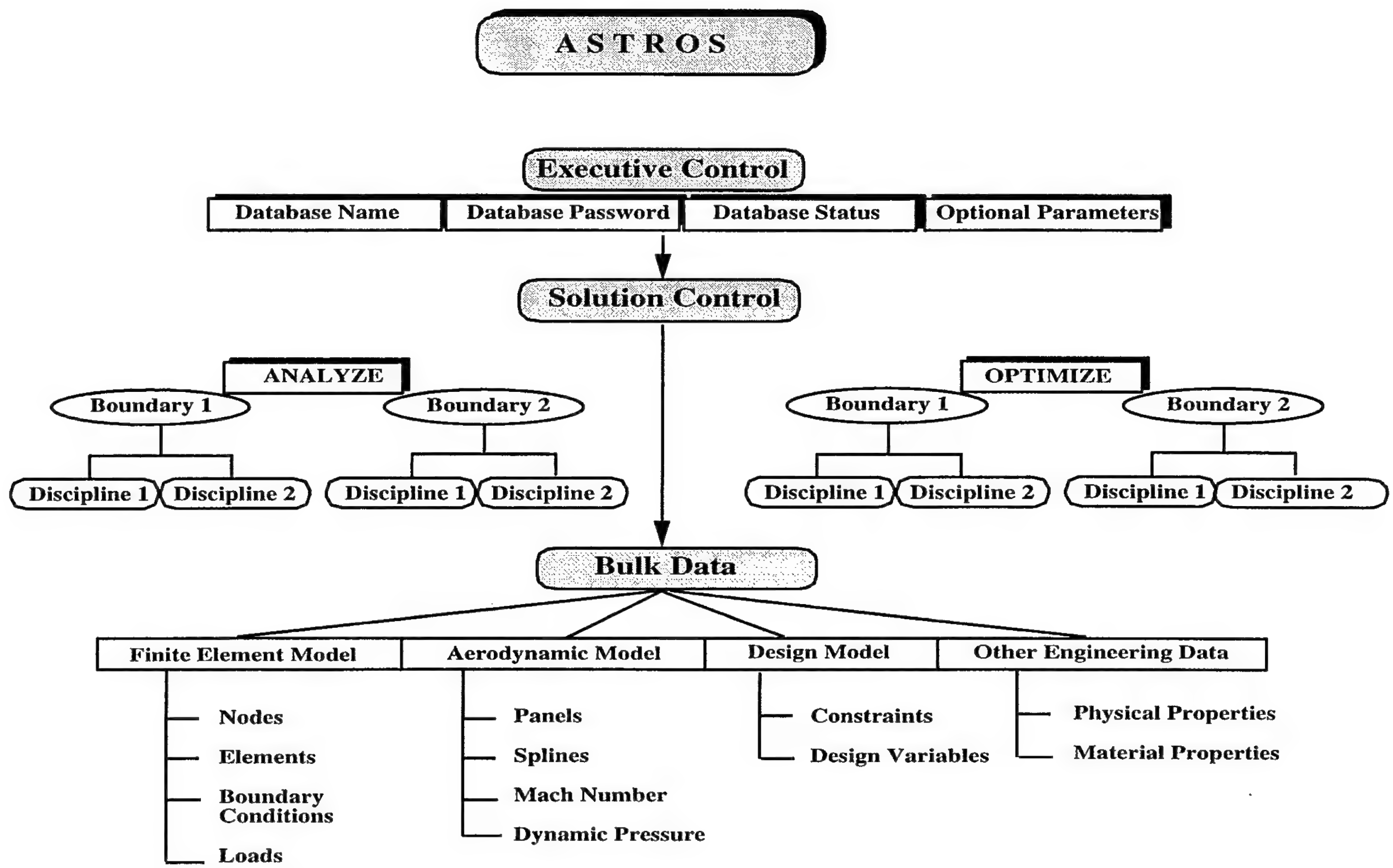


Figure 1. Hierarchy of ASTROS Input Model

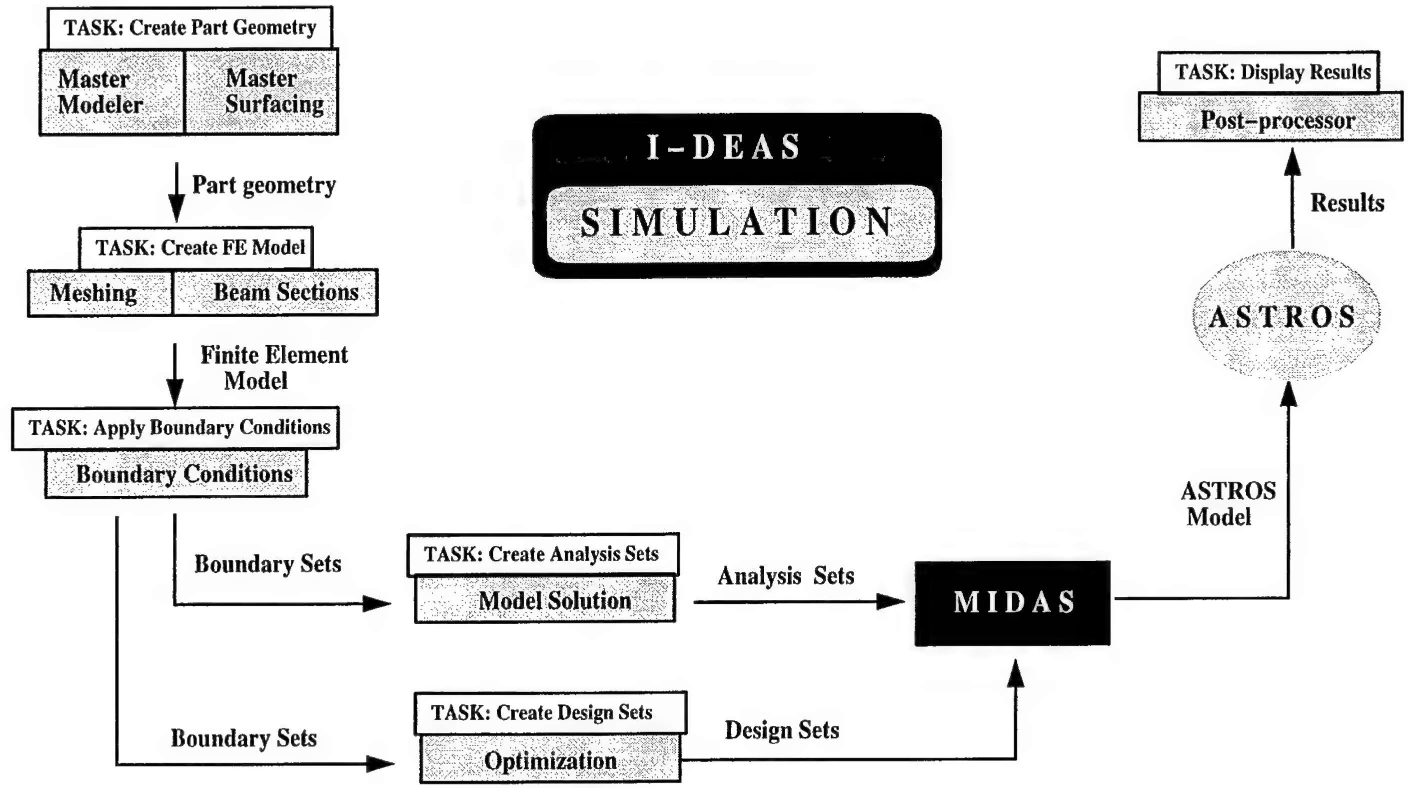


Figure 2. I–DEAS Simulation and its task functions

Chapter Three

MIDAS (Multidisciplinary Integrated Design and Analysis System)

MIDAS (Multidisciplinary Integrated Design and Analysis System) is an application developed for conducting preliminary parametric design studies on structures. It provides a platform to conduct iterative parametric preliminary design studies on the initial configuration of a structure until a satisfactory design is reached. It is a comprehensive tool developed to model a structure, design, analyze and evaluate its performance to ensure a safe structure. Modeling, analysis and design capabilities have been introduced into MIDAS through the interlinking of powerful structural modeling and analysis software available today. These software have been integrated on a common platform for MIDAS which provides the user a comprehensive environment to conduct design studies. Apart from providing a comprehensive platform for parametric design, MIDAS provides a powerful graphic environment for model generation. Model generation through MIDAS is mainly graphic in nature which scores over the text-oriented model generation methods in terms of its efficiency (speed, accuracy and easy modeling). The following sections describe the design, development and implementation aspects of MIDAS.

3.1 MIDAS Design

3.1.1 MIDAS Modeler (I-DEAS)

MIDAS has been designed as a shell that provides access to the structural modeling and analysis software. Since MIDAS was decided to have a predominantly graphic

mode of model generation, the next logical step was to decide on the most powerful and feasible graphic medium. The graphic medium chosen had to have the following characteristics:

1. Provide modeling, analysis and design features included in MIDAS

2. Provide capabilities for parametric modeling analysis and design

3. Essentially have a graphic mode of model generation

4. Provide a powerful and effective graphic user-interface

5. Provide features for integrating the external solvers

With these considerations, the I-DEAS Master Series software was chosen as the modeler for MIDAS. I-DEAS is one of the most powerful graphic software available today and has a very easy-to-use interface which allows easy, quick and efficient modeling of structures and also integration with external solvers. I-DEAS provides what is known as "Open Architecture" which lets the integration of external solvers and procedures. MIDAS has been designed with a lot of flexibility in order to enable the user to take full advantage of the I-DEAS graphic capabilities and use it to the fullest extent. Most of the CAD and finite element packages are stand-alone, and the user generally spends a lot of time and effort in transporting the CAD model and recreating a finite element model. In this process, sometimes complex geometry is very hard to recreate and could lead to modeling errors. I-DEAS Master Series has an integrated CAD-FEA environment which offers an advantage. Moreover this integration lends a seamless appearance to MIDAS, facilitating all the design tasks on a common platform, instead of the user having to resort to various software to design, analyze and evaluate results.

3.1.2 MIDAS Solver (ASTROS)

MIDAS is predominantly intended to provide graphic parametric design capabilities to the external solver ASTROS. Through "Open Architecture" tools mentioned in the preceding paragraph, MIDAS integrates I-DEAS with ASTROS. ASTROS is a tool for the preliminary design of components through finite element methods. It is a comprehensive software which provides a wide range of analysis and design capabilities not only for statics and modes disciplines but also for aerodynamics.

MIDAS essentially exploits the potentials of both the I-DEAS and ASTROS software, both its knowledge content and graphic facilities to the maximum extent, to provide a very easy to use platform to conduct design studies.

3.1.3 Advantage of using MIDAS over I-DEAS or ASTROS Separately

The user of ASTROS can benefit greatly from MIDAS, considering the fact that MIDAS makes up for the lack of graphic capabilities in ASTROS. It is definitely advantageous for a user to work with ASTROS through MIDAS considering the voluminous text input and output that ASTROS needs and generates. The structural modeling can be accomplished in much less time with greater efficiency with MIDAS than direct model generation for ASTROS. Also, it is more convenient to interpret results from ASTROS like mode shapes, stresses, etc, through a graphic display, rather than a text-oriented display.

3.1.4 Design Phases in MIDAS

The entire design cycle in MIDAS is divided into six stages:

1. Generation of the structural and/or design model

2. Transfer of the structural and/or design model to the solver

3. Solution phase wherein the actual computations take place

4. Retrieving the results of analysis and/or design through graphic means

5. Scanning the results

6. Redesign if necessary, by varying the required parameters

The six tasks mentioned above are grouped as : "pre-processing" (tasks 1 and 2), "solution phase" (task 3), "post-processing" (tasks 4 and 5) and "what-if" studies (task 6) in MIDAS as shown in Figure 3. Model generation using the I-DEAS modeler, model translation (I-DEAS format to ASTROS format) using MIDAS and transfer of the ASTROS model to the solver forms the "pre-processing" phase. The execution of ASTROS forms the "solution phase"; retrieving of results from ASTROS's binary database and evaluating the results through graphical means forms the "post-processing"; and redesign for varying parameters forms the "what-if" studies phase as shown in Figure 3.

3.1.5 Interconnection between the MIDAS Design Phases

The six stages mentioned above are executed in a cyclic manner as shown in Figure 3. The user of MIDAS develops the model in the "pre-processing" phase using the I-DEAS Simulation Pre-processor. The I-DEAS model is then transferred to MIDAS as shown in Figure 3, which translates it to a format compatible with ASTROS and transfers the model to it. The analysis or the design procedure is carried out within ASTROS in the "solution phase", which stores its results in a binary database (CADDB). The output results are then transferred from the database to MIDAS, converted to a suitable format for I-DEAS and then sent back to I-DEAS (as shown in Figure 3) where results are displayed graphically in the "post-processing" phase. If the design is satisfactory, it is frozen at this stage, else, the required parameters are

varied and the entire cycle is repeated as shown in Figure 3. This gives parametric design capabilities to MIDAS. This study enables what is known as the "what-if" design studies in which the user can study the response of the structure to variation in different parameters. Essentially there is flow of information from one phase of design to the other.

3.1.6 Pre-processing

Pre-processing in MIDAS involves developing the complete model of a component to be analyzed/designed using the external solver ASTROS. This forms the initial stage of design in MIDAS and is accomplished through the means of the graphic software I-DEAS and a model translator developed for this purpose. The various stages of pre-processing are illustrated in Figure 4. Pre-processing is carried out in two stages: first, developing the model graphically in the I-DEAS environment and second, translating or exporting this graphic model to a format suitable to ASTROS as shown in Figure 4. Using the I-DEAS Simulation tasks, the graphic model of the structure is first developed. The file translator developed for MIDAS is then activated, and the I-DEAS model information is passed to it as shown in Figure 4. The file translator receives I-DEAS model information and translates it to the ASTROS format and stores this information in a file. The end product of pre-processing is a finite element model ready to be delivered to ASTROS. This file is then passed to the ASTROS solver.

Stage one: After a design has been conceptualized, it is now ready to be brought in for a preliminary design of sizes and dimensions. The starting point for this would be generating a wireframe and solid model of the conceptual design. Pre-processing begins in MIDAS from the I-DEAS platform. I-DEAS has wireframe, solid and finite element modeling capabilities and is an icon driven software which simplifies modeling

tasks. A wireframe or solid model is first developed to match the overall size and dimensions of the conceptual model. A simple example of a wing geometry is shown in Figure 5a to illustrate the first stage of pre-processing. For a wireframe or solid model, the I-DEAS "Master Modeler" task is chosen which provides the necessary tools to construct the wing model (shown in Figure 5a). The model generated in this task is then brought in by the "Simulation" application of I-DEAS. In this application, the wireframe or solid model is transformed into a finite element model. For illustration purposes, a finite element model of the top skin of the wing is shown in Figure 5a. The "Simulation" application provides various tasks like "Meshing", "Boundary Conditions", "Model Solution", "Optimization", etc, for building the model. In the "Meshing" task, elements are generated, their physical properties like thickness or area defined, and materials are selected for the structure. In Figure 5a, the wing skin is meshed using quadrilateral elements. The wing skin thickness is specified as 0.2 in, and aluminum material is selected as shown in Figure 5a. In the "Boundary Conditions" task physical restraints are applied to the structure and also various load cases are defined and applied. Figure 5b shows the root of the wing cantilevered, and a distributed load of 500 lbs applied on the top skin. At this stage, any number of sets can be created for the model, although it is not necessary that all these be applied. The user can create analysis sets in the "Model Solution" task of I-DEAS shown in Figure 5b. The sets created in the "Model Solution" task will be used by MIDAS to generate cases for ASTROS.

An alternate method of pre-processing the model is from the "Simulation" task. If the user already has information concerning the nodal coordinates, the "Meshing" task of the "Simulation" application can be accessed directly, and the nodes can be generated. As an alternative to keying in these nodes, I-DEAS provides an option to import the finite element model through MSC/NASTRAN, COSMIC/NASTRAN,

ANSYS, etc. The ASTROS input largely matches with that of MSC/NASTRAN, and thus the user can use this import option to bring in an external model. Hence if an ASTROS file already exists with the finite element model information and the user wants to use this model, the MSC/NASTRAN import option can be used, instead of having to rebuild the model in I-DEAS. Currently for MIDAS, a capability is being provided for reading in an ASTROS file directly from the input file. This feature will be in addition to the already existing MSC/NASTRAN import option. In future, MIDAS will be able to read the model directly from the ASTROS database.

MIDAS uses the I-DEAS optimization capabilities to generate data for ASTROS. Using the "Optimization" task shown in Figure 5c, design variables (thickness, cross-section area) and design constraints (limits on stress, displacements and natural frequency of the structure) can be defined. Figure 5c shows the form for defining design variables. The elements are picked graphically to define design variables. Similarly Figure 5c shows another form to define constraints. Elements are picked graphically once again to define limits on them. Figure 5c shows a design variable being defined (thickness of the element) with a lower limit of 0.01, upper limit of 1.0 and initial value of 0.1. A stress limit of 25000 psi is imposed. This information is then translated to the ASTROS format.

Aero pre-processing in MIDAS is handled through the "Master Modeler" task. Since the aero module is unavailable in I-DEAS, a 'mapping' technique was developed along with an user-interface which enables the development of an aero model along with the structural model. Using the geometric modeling facilities (lines) in "Master Modeler" an aerodynamic model is developed according to the user's specifications. This technique will be discussed in detail in forthcoming sections.

Stage two: After a model is developed in I-DEAS, the next design stage in MIDAS is to translate or export this model to ASTROS. This required developing

a model translator in the I-DEAS environment which would read in the information from the I-DEAS model and write it suitably into the ASTROS format. This file translator was one of the key tools developed for MIDAS. It lays the basic foundation for execution of other tasks in MIDAS.

I-DEAS provides what is known as a 'model file' which contains the wireframe and/or solid and/or finite element model of the structure. Figure 6 shows the graphic model of the wing which in actuality is the I-DEAS model file. The I-DEAS model file is a database, which stores all the model information in different entities. The model translator accesses this database, picks up the required information and writes it into an ASTROS suitable format as shown in Figure 6. This is the principle for the development of the model translator.

The pre-processed information for ASTROS includes the executive control, solution control and bulk data information. The solution control includes references to multiple analysis types, multiple boundary conditions, multiple load cases and output options for each of these. The aerodynamic discipline and multiple design disciplines will be developed for the solution control in future work. Bulk data is further segregated into node and element, boundary conditions, load, physical property, material property, design variables, design constraints and aerodynamic model information. MIDAS extracts all this information one by one. Two options are provided to the user: either generate a completely new ASTROS model or modify an already generated model. To achieve this the pre-processing task is segregated into various parts, each performing a different task, i.e., one for node and element generation, one for boundary condition, one for load, one for solution control, one for design information, one for aerodynamic model information, etc. If the user is modifying a particular segment of the ASTROS model, say the boundary conditions, then only the boundary condition segment of the translator can be executed, and the ASTROS file will

update this information automatically.

Modifications to the ASTROS file can be made very easily. For example, if the user wants to apply a different set of boundary conditions and load sets, then the corresponding sets can be graphically picked in I-DEAS and applied on the model, and with the click of a button the modified ASTROS file can be generated effortlessly.

Once a satisfactory ASTROS model file is generated, the user can analyze or design the structure with ASTROS, through I-DEAS which forms the solution phase of MIDAS. It is not necessary for the user to exit I-DEAS and restart an ASTROS process. Instead, the user can simply click a button, and MIDAS prompts the user to name the file that needs to be submitted to ASTROS. When the file name is specified, MIDAS submits the file to the ASTROS process. Thus the user has no direct interaction with ASTROS. The entire process is carried out in a seamless manner within the I-DEAS environment.

3.1.7 Post-processing

Post-processing in MIDAS begins after the I-DEAS model has been translated to the ASTROS format and the analysis/design process executed. Post-processing in MIDAS includes retrieving the results of the ASTROS analysis/design and displaying them graphically on the I-DEAS platform. This phase can once again be divided into two stages: stage one being the extraction of results from the ASTROS database and stage two being displaying the results in I-DEAS. As mentioned before, ASTROS does not have graphic display capabilities. The user of ASTROS can use MIDAS to view and interpret ASTROS results in an easy and convenient manner.

Stage One: After the pre-processing and solution phase in which ASTROS is run, the results are ready to be retrieved and viewed as shown in Figure 7. ASTROS stores all the information from its model file in a binary database. Both input and

output information on the model are stored in this database in various entities. The first step in post-processing is to retrieve all the database information as shown in Figure 7 by the MIDAS post-processor. This was done by means of another file translator that was developed to open respective entities in the ASTROS database and extract information.

Stage Two: The information extracted from ASTROS now needs to be converted and stored in a format that is readable by the I-DEAS post-processor. This forms the second stage of MIDAS post-processing. A suitable translator was developed for this purpose. This translator picks up all the results extracted from ASTROS and stores them in appropriate I-DEAS model file data structures. These results are now available as any other I-DEAS result. Effectively, the ASTROS results are viewed through I-DEAS in the form of I-DEAS results. The results obtained are displayed using the I-DEAS graphic post-processing capabilities as shown in Figure 7.

The results retrieved so far include displacements, stresses, strain energy and mode shapes. Future work will include optimization and aero analysis results.

3.1.8 "What-if" Studies

The two design processes developed for MIDAS, namely pre-processing and post-processing make way for what is known as the "what-if" studies. Using the two features described above, the user can conduct a parametric study on a structure to obtain the design history. This information thus obtained can be used for two purposes: one is to study and determine the best design suitable for the purpose and the other is to study the response of the structure to parametric variations.

There are two forms of "what-if" studies to be incorporated into MIDAS. One form of study emerges from the already existing pre- and post-processing modules of MIDAS, and the other form of "what-if" studies will include maintaining a design

history to study structure response to parametric variations.

In the first form of "what-if" studies the user builds the model in I-DEAS as shown in Figure 8, which is then translated to the ASTROS format and solved for the requested quantities. For example, in Figure 8, the wing is analyzed for a force of 500 lbs acting on it. The results for this load case are then brought into I-DEAS through the post-processing module and displayed as shown in Figure 8. At this stage, the user can make a decision about the model, by studying the results. If the model response to a particular design parameter is not satisfactory, it can be modified either in the "Simulation" application or the "Master Modeler" task as the case may be, and the cycle can be repeated again with the pre- and post-processing. In Figure 8, the wing is shown to be re-analyzed for 300 lbs and the results re-evaluated. This form of "what-if" studies helps in deciding if the design is correct and satisfactory.

The other form of "what-if" studies is where the user wants to study the response variation to various quantities. This mode of "what-if" studies will be developed wherein a design history will be maintained for the requested quantities. For example, the user might be interested in studying the variation in the structure's weight with respect to the variations in the number of ribs or spars, with a constraint on the strength of the structure. In this case, for each set of ribs or spars defined, the optimum weight of the structure meeting certain strength requirements will be noted, and the variation in the structure's weight will be plotted against the number of ribs or spars.

3.2 MIDAS Development and Implementation

3.2.1 I-DEAS Open Architecture

As discussed in previous sections, I-DEAS is an integrated package capable of performing finite element analysis and design of structures for various disciplines. In

addition to offering these features, I-DEAS allows integration with other software packages. If a customer is using other finite element software which, unlike I-DEAS, may not have a graphic mode of model generation, this feature will prove to be of immense advantage. The user can generate the finite element model in I-DEAS and 'export' it to other packages. I-DEAS has existing data translators for popular software such as MSC/NASTRAN and ANSYS. For programs that are not supported currently, I-DEAS offers Open Architecture tools to customize I-DEAS to any external software. Since ASTROS is not one of the software supported by I-DEAS, the Open Architecture tools were used to customize I-DEAS for an ASTROS input stream.

Before deciding on I-DEAS Open Architecture, several other options were considered. One of them was the I-DEAS universal file. When engineers at Structural Dynamics Research Corporation (SDRC) were consulted, it was learned that universal files would not be supported in the future releases of I-DEAS. The other choice was a dataloader which was not recommended as a good tool. Considering all these drawbacks, the I-DEAS Open Architecture proved to be the best choice.

The I-DEAS Open Architecture is an open system environment which allows user-interface customization, direct data access and direct command and function access. It allows the programmer to customize, automate and extend the standard capabilities of I-DEAS to other applications which in this case is ASTROS.

The Open Architecture offers standard, product specific and optional tools for user-interface and customization. Standard components that come integral with the Open Architecture are Open Language, Open User Interface and Relational Data Interface. Product Specific tools are Open Solve, Open Test and Open Drafting. Optional tools include Open Link, Open Data and Open Advisor.

Open Language: Open Language lets the user capture I-DEAS sessions using interface based command mnemonics and tailor them for multiple executions. It can

also be used for custom prompts in the I-DEAS prompt window. Open Language allows definition of global symbols to execute I-DEAS commands or external files from within the I-DEAS software.

Open User Interface: Open User Interface allows the programmer to customize I-DEAS user-interface in several ways. Options for colors, fonts, etc can be set with I-DEAS; application and task level menus can be modified or created. Also existing icon panels can be modified or new icons created.

Relational Data: It is a stand-alone relational tool which allows access and creation of relational data outside of I-DEAS. The Relational Data Application Programmer Interface allows development of applications that require relational data storage and management.

Open Solve: It allows customized operations in the I-DEAS solver to perform solutions specific to the user's needs. It accesses model databases, matrix databases and universal files to read and write matrix and solution data.

Open Test: It provides Test Data Access for the I-DEAS Test Engineer and allows the programmer to write interfaces, access associated data files, manage communications and support a wide range of existing interfaces

Open Drafting: The I-DEAS Drafting Programming Language supports interface based command, macros and a robust suite of functions for direct access to and modification of the drawing database.

Open Link: Open Link provides command and function access to the I-DEAS model database for integrating custom and third party applications with I-DEAS.

Open Data: Open Data is an Application Programmers Interface (API) which provides direct access to the I-DEAS database unlike Open Link. Finite element models and results can be accessed directly using this feature.

Open Advisor: It provides access to the I-DEAS Analysis Advisor development

environment and allows inclusion of user-defined rules to the knowledge base within the I-DEAS finite element software.

The choice of Open Architecture tools from the above mentioned list for MIDAS was based upon the features each of the above offered. The primary requirement for MIDAS is a tool that can access I-DEAS information. From a survey of all the Open Architecture tools it can be seen that the most useful ones would be Open Link, Open Data, Open Language and Open User Interface. Figure 9 shows how Open Architecture is used in MIDAS. Using Open Language (for defining commands to run the file translator) MIDAS uses Open Link and Open Data for information access from the I-DEAS model database. Since Open Link and Open Data are used to query model information and I-DEAS is used to reply to these queries, Open Link and Open Data are said to act in a 'client' mode and I-DEAS in a 'server' mode. Open User Interface is used for developing a front end for user-interaction with MIDAS as shown in Figure 9 for querying and getting information from the user. Each of these Open Architecture tools are described briefly here.

3.2.1.1 I-DEAS Open Architecture Tools in MIDAS

Open Link: I-DEAS Open Link software provides the Command-Reply System Application Programming Interface that lets integration of third party supplied programs with I-DEAS. Open Link programs are executed in the I-DEAS environment and can be started using user-defined icons or from the I-DEAS prompt region. An Open Link application can generate input for another program, such as an analysis code. Open Link, however does not provide direct access to the I-DEAS model database. It forms an intermediary between I-DEAS and the third party solver. The basics of using Open Link for data access include using the I-DEAS command

mnemonics. The mode of data extraction will be described in forthcoming chapters.

Open Data: Central to Open Data is the Application Programmers Interface that provides direct access, unlike Open Link, to data stored in the I-DEAS model file for custom data translation and management. C applications can be written using functions provided by Open Data. Custom applications developed with Open Data can also take advantage of I-DEAS Open Link software to interact directly with an I-DEAS session. Open Link allows you not only to access geometry created in other applications, but also to interact with an I-DEAS session by sending commands and retrieving results. Open Data provides the Geometry Application Programmer Interface for accessing geometry in the I-DEAS model file and also the Finite Element Modeling Application Programmer Interface for accessing Finite Element Modeling (FEMAPI) data. Using FEMAPI functions, applications can be written to create, query and delete data. MIDAS predominantly uses FEMAPI functions to query and create data for pre-processing and post-processing, respectively.

Open Language: I-DEAS Open Language offers an alternative to direct user-interaction with the I-DEAS software. Open Language offers several features such as global symbols, program file, variables, etc. MIDAS uses global symbols to run Open Link or Open Data programs. Global symbols are user-defined synonyms that define a name that is replaced by the associated string when you enter the global symbol name. Command strings to run external programs can also be associated with it. In MIDAS, global symbols will facilitate interacting with ASTROS in the background within I-DEAS, without the user having to move between different platforms.

Open User Interface: Open User Interface offers customizing the I-DEAS menu, adding applications/task menus, adding new/modifying icons, etc. MIDAS will use the adding applications and task menus and the adding/modifying icons features. A front end for MIDAS was developed using this feature of the I-DEAS

Open Architecture. This tool will be used to interact with users directly to get certain inputs.

3.2.2 ASTROS Model Information in MIDAS

A typical model of ASTROS extracted by MIDAS from I-DEAS has the following data segments:

1. Executive Control
2. Solution Control
3. Bulk Data

3.2.2.1 Executive Control:

The ASTROS user directs the system through an input data stream composed of a command to attach the ASTROS runtime database. The first entry of this ASTROS command is "ASSIGN DATABASE" as shown in Figure 10. This enables the user to attach the runtime database file (TEST) that is used during ASTROS execution. It is in this database that ASTROS stores all the model information, both input and output. This binary database is useful in the post-processing module of MIDAS, where all the information is read, translated and displayed. A user can create many databases for a single I-DEAS model file and maintain a history of all the modifications in the model. The following quantities pertaining to Executive Control shown in Figure 10 are extracted: a. Runtime Database Filename b. Database Password c. Database Status d. Optional Parameters.

3.2.2.2 Solution Control

The ASTROS modules considered for MIDAS include Linear Statics and Normal

Modes (STATICS and MODES), Aerodynamic Analysis and Design (SAERO and FLUTTER). The present work developed the solution control for the Linear Statics and Normal Modes analysis disciplines as shown in Figure 10. Future work will include the solution control for optimization and also solution control for aero analysis disciplines. The information for Solution Control was obtained from the I-DEAS model file. It must be mentioned here that the solution control information required for ASTROS was not directly obtainable from I-DEAS. The I-DEAS model file had to be read and appropriate inferences had to be made in order to construct the ASTROS Solution Control packet.

The information extracted for the Solution Control included boundary conditions, loads, disciplines and output information as indicated in Figure 10. The boundary condition information consists of the physical restraints, constraints and degrees of freedom specified for the structure. The corresponding ASTROS translations are indicated in Figure 10. I-DEAS provides options to create physical restraint sets which can be translated to single point constraints (SPC) in ASTROS. Constraint sets can be created in I-DEAS which translates to multi-point constraints (MPC) in ASTROS. I-DEAS also provides options to create a degree of freedom set in which the degrees of freedom that need to be omitted (REDUCE) from the defined problem are defined. To write the boundary conditions of Modal analysis (MODES), MIDAS scans the model for any boundary sets with the Modal analysis discipline specified. If any such sets are found, MIDAS includes this component of the boundary conditions along with the others. Dynamic reduction control data (DYNRED) for modal analysis is another boundary condition of ASTROS, that was added in MIDAS.

The load types considered for MIDAS include mechanical (MECH), thermal (THERMAL) and gravity (GRAV). ASTROS can handle all these load types simultaneously or in any combination. I-DEAS provides the user the option to create

mechanical, thermal and gravity load sets. Multiple load sets for each boundary condition is enabled in both I-DEAS and ASTROS.

I-DEAS "Model Solution" has the output option specifications. The outputs in I-DEAS include displacements (DISP), stresses (STRESS), strain (STRAIN), strain energy (ENER), element forces (FORCE), and mode shapes (DISP, ROOT). Each of these has three options, either to 'Store' or 'Store and List' or 'No Output'. When MIDAS encounters the first two options, it prints these quantities and when it encounters 'No Output', it suppresses the printouts.

ASTROS can handle multiple boundary conditions and multiple disciplines for analysis and design. Currently, the Linear Static analysis and optimization, Normal Modes analysis and optimization, Unsteady Aero analysis disciplines have been considered. Future work will include unsteady aero optimization and steady aero analysis and optimization. MIDAS was tailored to interpret the I-DEAS analysis sets to add multiple boundary conditions and/or disciplines feature. Figure 11 illustrates this. Two analysis sets are created in the Model Solution task of I-DEAS, and each of these sets has a boundary set assigned as shown in Figure 11. Boundary set 1 has restraint set 1 and constraint set 1 applied. Linear Statics is the discipline defined for which there are two load cases as indicated in the figure. Boundary set 2 has restraint set 2, and the modal analysis discipline is specified. Although I-DEAS treats all these sets as individual problems, MIDAS interprets them as all the sets belonging to a single solution set in ASTROS. When translated, the ASTROS file generated looks as shown in Figure 11. In this file, a single analysis set is specified which has two boundary conditions (BOUNDARY SPC = 1, MPC = 1 and BOUNDARY SPC = 2) and two disciplines (STATICS and MODES). This way, the user can introduce multiple boundary conditions and/or multiple disciplines that need to go into ASTROS within I-DEAS itself.

3.2.2.3 Bulk Data

Each parameter in the solution control references a corresponding entry in the bulk data deck in ASTROS. Data extracted for Bulk Data correspond to the ones extracted in the Solution Control. Bulk data information extracted from I-DEAS and translated to ASTROS is indicated in Figure 12. Coordinates for the grid points (GRID) were first extracted. Then the element connectivity (CQUAD, CROD, etc) was read from the model file. The element types considered for bulk data included all the elements present in the ASTROS library. They included one dimensional, two dimensional, three dimensional and other scalar elements. The boundary conditions corresponding to the ones extracted for the solution control were written as shown in Figure 12. They included single point constraints (SPC) consisting of all the nodes that were physically restrained along with the degrees of freedom restrained. The multipoint constraint (MPC) included the independent and dependent nodes and their coefficients. The degrees of freedom that were to be omitted (OMIT) were extracted and also the dynamic reduction control data (DYNRED) and modal analysis parameters (EIGR).

Mechanical, thermal (TEMP) and gravity (GRAV) load magnitudes along with the location at which they act were extracted for all the load cases. Both forces and moments (FORCE, MOMENT) were extracted for the mechanical loads.

Physical properties such as the cross-section area, thickness, moments of inertia, etc, were extracted for various element types (PSHELL, PROD, etc). Also the material specifications (MAT1) such as the young's modulus, shear modulus, density, poisson's ratio, etc, were extracted.

Information on design constraints on displacements (DCONDSP), Von Mises stress constraint on elements (DCONVM), material and physical property identi-

fication (DCONVMM, DCONVMP) and natural frequency constraints (DCONFRQ) were extracted. Design variable information for elements with unique linking and with physical linking (DESELM, DESVARP) were extracted. User defined optimization parameters (MPPARM) to override default values were also extracted.

Information extracted for the aerodynamic module include aerodynamic physical data such as reference density and reference length (AERO), aerodynamic panel element connection for unsteady aero (CAERO1) and aerodynamic lists for division points of chordwise and spanwise boxes (AEFACT).

Here again, in the bulk data packet, data was extracted for multiple boundary conditions, multiple disciplines and multiple load cases.

3.2.3 I-DEAS to ASTROS Model Translation Methodology

MIDAS extracts information for the ASTROS model from the I-DEAS database using Open Architecture tools. Different Open Architecture tools were used either individually or in combination according to the input module being considered. The four Open Architecture software that were used are: Open Link, Open Data, Open Language and Open User Interface.

A general procedure for extracting the information is described as follows.

3.2.3.1 Use of Open Link for I-DEAS to ASTROS Translation

MIDAS uses Open Link and I-DEAS in a client-server mode. In MIDAS, Open Link is the client and I-DEAS is the server. Open Link in client mode, queries I-DEAS for information through I-DEAS command mnemonics, and I-DEAS in a server mode, responds to these queries by sending back results. An example of data extraction using I-DEAS command mnemonic is presented here:

Figure 13 shows MIDAS querying the I-DEAS model database for information

on a boundary condition set. To query, MIDAS sends the following command to I-DEAS through Open Link: '/B OP BC ? 1 #DUMP' as shown in Figure 13. All commands begin with a '/'. Each entry of the mnemonic represents one particular parameter. For example, 'B' is querying the "Boundary Condition" task. 'OP' refers to operations to be performed in this task. 'BC ? 1' refers to querying the first boundary condition set. Any boundary condition set that needs to be queried can be substituted here in place of '1'. '#DUMP' dumps all the information about this boundary set. Using this command, Open Link queries the I-DEAS model database as shown in Figure 13, and I-DEAS gives out the result for the query as indicated in Figure 13.

From I-DEAS result, information for ASTROS can now be obtained as follows: Consider the 2nd line of the result which is 'BT: MNU 1' (in Figure 13). The MNU number 1 refers to linear static analysis in I-DEAS. 'OP_BC: NND 1 1' says that there is only one boundary set applied to the problem and the set number is 1. In the next and subsequent lines, the restraint set (SE_RT: TOG 1), and the constraint set (SE_CT: TOG 1) are applied since their toggles are set to 1. No kinematic degree of freedom set is applied since its toggle is set to 0 (SE_KT: TOG 0). The restraint set number is denoted as 1 (SE_RES: NND 1 1), the constraint set number as 1 (SE_CO: NND 1 1) and the kinematic degree of freedom set as 0 (SE_KI: NND 1 0). Considering the loads acting, the result indicates that thermal load set number 2 is applied (SE_TS: NND 1 2) along with 2 mechanical/gravity load sets (SE_LO: DMN 2 1 2). The two numbers 1 and 2 indicate that load set numbers 1 and 2 are applied.

This information obtained from I-DEAS is now translated to ASTROS format shown in Figure 13. This ASTROS translation can be explained as follows: For the same restraint (1) and constraint set (1) there are two load cases: 1 and 2. Thermal loading is used in both sets, since it belongs to the boundary set as a whole. Hence

thermal set 2 is applied along with mechanical and gravity set 1 for load case 1, and thermal set 2 is applied along with mechanical set 2 for load case 2. (Here it is assumed that the load set one contains mechanical and gravity loading and the load set two contains only mechanical loading). The above example also illustrates multiple load cases (two STATICS cases). Consider an example of multiple boundary conditions and multiple disciplines. Figure 14 shows two boundary sets created in MIDAS which are queried one after the other. The results of this query are interpreted as shown in Figure 14. This case has two boundary sets (BOUNDARY SPC = 1, MPC = 1 and BOUNDARY SPC = 2) and two disciplines (STATICS and MODES).

3.2.3.2 Use of Open Data for I-DEAS to ASTROS Translation

Open Data provides the functions, which MIDAS uses to read the finite element data from the I-DEAS database as shown in Figure 15. Each segment of the finite element data is associated with an Open Data function. This function accesses a specific data structure containing all the information. For example, in Figure 15, MIDAS is querying the I-DEAS database for single point constraint information using the Open Data FEM service. The function used for this purpose is 'oa_QueryNodalDispRest'. Arguments like the node label, restraint set number and the I-DEAS model identification number are passed to this function as inputs shown in Figure 15. Open Data queries the I-DEAS database and gets results which include the degrees of freedom restrained. It is to be noted here that this information is not printed directly. The function 'oa_QueryNodalDispRest' returns a 'degree of freedom mask', which is translated to the appropriate degrees of freedom. The function can return any value between 0 and 63. This number is converted to a binary number and appropriate conclusions are made as to which degree of freedom is restrained. ASTROS denotes the degrees of freedom in the following manner: 1 for X translation, 2 for Y translation,

3 for Z translation, 4 for X rotation, 5 for Y rotation and 6 for Z rotation

The function sent to Open Data to query will look like:

oa_QueryNodalDispRest(10,100,2,iTimeFlag,dTime,&NodResInfo)

where 10 is the model ID, 100 is the node number, 2 is the restraint set number, iTimeFlag is set to 0, dTime is the time variation value and &NodResInfo is the data structure containing all the nodal restraint information. From this structure, the item of our interest NodResInfo.DofMask; denoting the degree of freedom mask is extracted, and the corresponding single point constraint card (SPC) is written in the ASTROS format shown in Figure 15. The second field denotes the single point restraint set number. The third field indicates all the degrees of freedom that are restrained for that set. The subsequent fields indicate the grid point numbers that are restrained.

3.2.3.3 Use of Open Language for I-DEAS to ASTROS Translation

Global symbols that form part of I-DEAS Open Language are used to execute segments of MIDAS. The input model in ASTROS is generated either as a whole or in segments. In either case, global symbols associated with an I-DEAS Open Architecture command are used. For example, the command "oaxx run OA_BC" is the command defined in MIDAS to run the boundary condition segment of the ASTROS input. Instead of having the user use this all the time, it can be associated with a global symbol which can be just one word, say, 'boundary'. So when this symbol 'boundary' is entered at the I-DEAS prompt, the command "oaxx run OA_BC" is executed.

3.2.3.4 Use of Open User Interface for I-DEAS to ASTROS Translation

There are some segments of input in ASTROS which are unavailable in I-DEAS.

This resulted in using other means of data extraction from the user. One of the simplest ways is to provide a graphic user-interface for inputting the data. As an example, consider generating modal analysis parameters, for which I-DEAS does not provide all the parameters required for ASTROS. A user-interface shown in Figure 16a was thus developed to interact with the user to get the necessary information. The user-interface seen in the figure is a graphic-based application in the X window environment using the Open Software Foundation (OSF)'s Motif toolkit. The industry standard X window system adopted consists of Xlib and XtIntrinsic functions to make the dialog window, push-buttons and text widgets shown in the figure. Five push-buttons are provided in the window to make a selection of the modal analysis method as shown in Figure 16a. When one of the methods is selected by clicking on it, MIDAS notes down this method. The user is further prompted to enter information like the frequency range, number of roots to be printed, etc, as shown in Figure 16a. Text widgets are provided wherein the data can be keyed in. The "OK" push button shown here is associated with a callback which notes all the information entered in the file. On entering this information and clicking on "OK", MIDAS invokes the callback procedure and translates the modal analysis parameters to the ASTROS format shown in Figure 16a. Here, the Given's method of modal analysis is selected (GIV). The frequency range of interest is specified to be 0 to 12, the estimated number of roots is 12 and 3 are to be printed. Dynamic reduction control is specified (shaded YES in Figure 16a), and the mass conversion factor (CONVERT MASS) is given as 2.59E-3.

On similar lines, Figure 16b shows the interface developed for the ASTROS executive control information extraction using the Open User Interface. Four text widgets are provided to enter the executive control parameters. After entering the information, clicking on 'OK' writes the executive control information shown in Figure 16b.

The four Open Architecture modules mentioned above were used in combination in certain segments. Although Open Data was the primary choice because of its directness, it was not possible to use this all the time, because some information could not be obtained using Open Data alone. A combination of Open Data and Open Link worked best at some places. Extraction of data using Open Data is considerably faster, and it has been preferred over Open Link whenever it was feasible to use it. For example, for extracting nodal restraints, although Open Data provides this information, it is not possible to query the restraint sets that were actually created. This information had to be obtained from Open Link. Given this information, Open Data is capable of extracting all the restraint data.

3.2.3.5 Executive Control Extraction

The extraction of executive control data using the Open User Interface has been discussed in this chapter in the context of explaining the Open User Interface for MIDAS.

3.2.3.6 Solution Control Extraction

The extraction of solution control data has been discussed in this chapter in the context of using Open Link for MIDAS.

3.2.3.7 Node and Element Extraction

Node and element extraction was treated as one segment in MIDAS due to its close connectivity. This segment of data was extracted using both Open Link and Open Data. I-DEAS lets the user generate elements either through manual picking or through mesh generation. Nodes and elements are first generated, and the corre-

sponding segment for ASTROS is activated as shown in Figure 17 by clicking on 'Node and Elements'. MIDAS uses Open Link to query the number of nodes/elements and their labels and sends this result to Open Data to query the node/element information such as nodal coordinates, element type, element connectivity, etc, as shown in Figure 17. I-DEAS uses numbers to denote element types. MIDAS reads this number and makes the required conclusions regarding the element. For example, in Figure 17, the I-DEAS element type is 94, which belongs to the thin shell linear quadrilateral element type and MIDAS writes it as CQUAD4 for ASTROS. Each element is assigned a physical property card. The physical property could refer either to thickness of the element or cross-section area of the element or moments of inertia, etc. In I-DEAS, before the creation of elements, their respective physical property tables are created. These are then assigned to the elements. MIDAS uses this physical property table number to denote the physical property card in ASTROS.

The nodes and elements defined in I-DEAS are translated by MIDAS into suitable formats in ASTROS. In Figure 17, the second field of the GRID card denotes the grid point number and the remaining fields, the X, Y and Z coordinates. The first field of the element card denotes the element type through a keyword (CQUAD4 in this case). The next field refers to the element label. The third field references the property table associated with the element. The remaining fields denote the element connectivity. This general format is followed for all the element types in ASTROS.

The element types extracted by MIDAS for ASTROS include:

1. Thin shell elements: Both triangular and quadrilateral (CTRIA3, CQUAD4)
2. Membrane elements: Both triangular and quadrilateral (CTRMEM, CQDMEM1)
3. Plane stress elements: Quadrilateral (CSHEAR)
4. Beam elements: Linear beam (CBAR)
5. Rod elements: (CROD)

6. Solid elements: Linear, parabolic and cubic solid (CIHEX1, CIHEX2, CIHEX3)

7. Lumped mass: (CONM2)

8. Translational and Rotational Scalar spring elements: (CELAS2)

All the elements card syntax are presented in the example section of this work. One of the principle issues addressed during the extraction of elements was establishing compatibility between element types in ASTROS and I-DEAS. Compared to ASTROS, I-DEAS has a wider range of elements and it was essential to map the elements in I-DEAS to ASTROS. For this, the characteristics of the ASTROS elements were studied along with those of the I-DEAS elements. The elements of I-DEAS whose characteristics matched that of ASTROS were listed against ASTROS elements. This established a suitable compatibility between the elements in ASTROS and I-DEAS. Figure 18 shows the element mapping done for I-DEAS and ASTROS. The corresponding physical property types for the elements are indicated against the ASTROS element type in Figure 18.

3.2.3.8 Boundary Condition Extraction

The boundary conditions extracted in MIDAS include single point constraints, multi-point constraints, kinematic degrees of freedom, dynamic reduction control and modal analysis method references. Here again, Open Link and Open data were used in combination with Open User Interface for data that was not available in I-DEAS.

Single point constraints (SPC): The extraction of single point constraint data has been explained in the context of data extraction using Open Data. Single point constraints extracted for MIDAS include the following types of physical restraints:

1. Clamp joints

2. Ball joints

3. Pin joints
4. Slider joint
5. Roller joints
6. User specified

MIDAS is capable of extracting data for all the above mentioned physical restraints both when applied individually or in combination .

Multi-point constraints (MPC): Multi-point constraint information was extracted using Open Link functions. Multi-point constraints enforce a relationship between the nodal degrees of freedom of different nodes. A multi-point constraint (MPC) equation is a linear algebraic equation which has nodal displacements or rotations as unknowns. It is used to model the condition where the linear combination of one or more nodal displacements or rotations must be equal to a constant or zero. It is expressed as:

$$A_j U_j = Z_j$$

where j is number of the nodal displacements to constrain, Aj is any non-zero real number (in the matrix of constraint coefficients), Uj is nodal displacement or rotation which are the unknowns in the equation and Zj is any real number (could also be zero).

A constraint set is created which can include multiple MPC equations. For example, Figure 19 shows constraint set 1. Constraint equations will be placed in this set. In the wing model, grids 111, 112, 121 and 122 have concentrated masses attached to them, but no structure. Instead, multipoint constraints are used to make the motion of these grids dependent on the motion of the structural beams representing the wing elastic axis that extends from grids 100 to grid 110 and 120.

The MPC equations can be created for ASTROS, using the form shown in Fig-

ure 19. The dependent and independent nodes can be graphically picked and the coefficient and constant term values can be entered as shown in Figure 19. Once the equations have been defined, the boundary condition segment of MIDAS is activated which reads in the equations and writes them to the ASTROS format as shown in Figure 19.

The second field of the MPC card denotes the constraint set number. The third field refers to the dependent node, the fourth field to its active degree of freedom and the fifth to the coefficient. The consecutive three fields refer to the independent node, its active degree of freedom and the coefficient.

Thus the equations can be generated in a very convenient manner through graphical picking rather than the conventional text-oriented manner.

Kinematic degree of freedom set: A kinematic degree of freedom set can be defined in which all the degrees of freedom which the user wishes to omit from the problem through matrix partitioning can be included. Through the form shown in Figure 20 the user selects the active and inactive degrees of freedom for a set of nodes. When the node selection is made, a form shown in Figure 20 is brought up in which the active and inactive degrees of freedom can be specified simply by clicking. A space truss is shown in the figure which needs to be designed for optimum weight. It has been modeled with CROD elements. Kinematic degrees of freedom can be defined for the structure to make the translations of some nodes inactive (eg. for nodes 1,2,5,7, etc)

This information is read by MIDAS through Open Link functions and translated to ASTROS format as shown in Figure 20.

The second field denotes the kinematic degree of freedom set, the third field the node number for which the degrees of freedom are reduced and the fourth field refers to the components of translations or rotations that are reduced. Three such nodes

can be referred to on an OMIT entry.

3.2.3.9 Load Sets

Three types of load sets, namely, mechanical, gravity and thermal are included in MIDAS. These load sets act either individually or in combination.

To extract the load set data, each boundary condition set is scanned for the load sets contained in them and each of them is then queried individually to determine the type and magnitude of the loads. Mechanical and gravity load sets can be created either in individual sets or can be placed in a common set. Hence MIDAS scans for mechanical and gravity loads simultaneously. Thermal load sets are created and applied individually to the boundary sets.

A mechanical load set can be created by graphically picking the grid points and assigning a magnitude for the load using the load set form shown in Figure 21. A number of such sets can be created. A gravity load can be included along with this set by graphically picking the gravity vector on the structure. When the load set is queried using Open Link, the result obtained contains information about the load type and magnitude. This mechanical load information (FORCE) is translated to ASTROS as shown in Figure 21.

The second field denotes the load set number, the third field the grid point on which the load acts, the fifth field denotes the magnitude of the force and the remaining fields denote the components of the vector, which signifies the direction in which the force acts.

Along with mechanical forces, moments can also be defined using the form shown in Figure 21. The syntax of the MOMENT card is similar to the FORCE card except that the fifth field denotes the magnitude of the moment.

Gravity vector components can be created in MIDAS by graphically selecting the direction of the gravity vector on the structure and entering the vector magnitude at the I-DEAS prompt as shown in Figure 21. When MIDAS encounters a gravity vector, it queries the set further for the gravity vector components. These components are used in ASTROS to determine the gravity loading. The gravity vector components are read off and written to ASTROS as shown in Figure 21.

The second field signifies the set identification number, the third field refers to the coordinate system identification number, the fourth field signifies the gravity vector scale factor and the remaining refer to the gravity vector components.

Thermal load sets are defined separately from the mechanical and gravity load sets. Temperatures can be defined for grid points, again through graphical picking. A set of nodes which are at the same temperature are picked and using the form shown in Figure 21, the element type is first selected, whether Solid/Plane2D or thin shell type and the amplitude of the temperature is defined. This information is scanned by MIDAS and translated to the ASTROS format. For example, Figure 21 shows a structure in which the grid points are at 200^oF. The temperature information translated to ASTROS format is shown in Figure 21.

The second field denotes the temperature set, the third field the grid point and the fourth the amplitude of the temperature. Three such entries can be included in one TEMP card. ASTROS uses this temperature amplitude to compute thermal loading on the structure.

3.2.3.10 Physical Properties

The physical properties of an element refer to thickness, cross-section area, area moments of inertia, torsional constants, etc. Physical property tables are first created

for different element types, and each element is associated with a particular physical property entry. While querying the element, its respective physical property reference is noted for further queries. Once again, using a combination of Open Link and Open Data, all the data pertaining to the physical property of the element are extracted. A physical property table number is supplied to I-DEAS which gives the result containing all the information.

Properties were extracted for the following element types:

1. Thin shell elements: Both triangular and quadrilateral (PSHELL)
2. Membrane elements: Both triangular and quadrilateral (PTRMEM, PQDMEM1)
3. Plane stress elements: Quadrilateral (PSHEAR)
4. Beam elements: Linear beam (PBAR)
5. Rod elements: (PROD)
6. Solid elements: Linear, parabolic and cubic solid (PIHEX1 for all)

3.2.3.11 Material Properties

Material properties were extracted using Open Link functions. Materials present in the material database (that were used in the structure) were queried for the following parameters:

1. modulus of elasticity
2. poisson's ratio
3. mass density
4. shear modulus
5. coefficient of thermal expansion
6. thermal expansion reference temperature

These properties were written in the following ASTROS format:

MAT1 2 1.0E5 0.3 0.1 6.5E-6 5.3E-6

Field two refers to the material identification number. The third field refers to modulus of elasticity, the fourth to the shear modulus (ASTROS calculates this value, hence left blank), the fifth to poisson's ratio, the sixth to mass density, the seventh to coefficient of thermal expansion and the eighth to the thermal expansion at reference temperature.

3.2.3.12 Modal Analysis Data

The extraction of modal analysis data using the Open User Interface was discussed in this chapter while explaining the Open User Interface for MIDAS.

3.2.3.13 Design Variable Data

Design variable information (element thickness or cross-section area) was extracted for all the ASTROS element types. Design variable information was extracted using the Open Link and Open User Interface. The current version of Open Data does not provide access to optimization data in the I-DEAS model. Properties have been defined for design variables that are uniquely associated with a single finite element (DESELM) and also for physically linked global design variable properties (DESVARP). Two types of design variable linking have been enabled in MIDAS; one, linking the elements directly and two, linking the elements through their physical property. The first form of element linking has been done through the I-DEAS Open Link and the second form through the Open User Interface.

Figure 22 shows a simple rectangular wing. The three ways of defining design variables in MIDAS are shown here. The first way is to consider individual elements as design variables. For this the elements are picked graphically and for each of

these elements, the upper limit, lower limit and initial value are defined as shown in Figure 22. For example, elements 13 and 14 are picked individually. The ASTROS translation for this type of design variable definition is presented in Figure 22. The second field refers to the design variable number, the third field refers to its associated element number, the fourth field refers to the element type and the remaining three fields specify the lower limit, upper limit and the initial value of the design variable.

To define the design variables in the physical linking format, all the elements that need to be linked are picked one after the other as shown in Figure 23, and the upper limit, lower limit and initial value are defined as shown in Figure 23. In this figure, elements 20, 26, 13, 17, 21, 23, 14 and 16 which form the elements on the wing skin are linked and defined as the first variable. Elements 24, 27, 11, 25, 26 and 12 are linked (not shown graphically here for the sake of clarity, but the ASTROS translation is shown in the figure) and defined as a second design variable. Similarly other elements can be linked if necessary. The translated information for this type of element linking is shown in Figure 23. The design variable card refers to an element list (ELIST) which lists all the elements that were linked. The second field of DESVARP refers to the design variable number, the third field refers to the ELIST card and the remaining three fields specify the lower limit, upper limit and initial value of the design variable. The second field of ELIST entry refers to the list identifier, the third to the element type and the remaining fields refer to the elements that have been linked.

To link design variables through their physical property in physical linking, the following methodology is followed: ASTROS allows both direct element linking and physical property linking simultaneously. MIDAS first lets the user link the elements directly. After this is done, it scans for the physical property cards which have not been used in direct element linking. Some physical property card numbers are selected and presented for further physical property linking. This is shown in Figure 24. This

method eliminates the possibility of any error on the part of the user in duplicating element linking. From the form shown in Figure 24, the user can select the list of property cards whose corresponding elements need to be linked. This information translated into the ASTROS format is shown in Figure 24. This design variable card refers to a property list (PLIST) that lists the identification number of all the physical property cards that were linked. In this way, all the elements, that referred to these physical property entries are linked. This is an indirect way of linking elements and can be preferred over direct element linking to save the number of lines of input. The syntax of the DESVARP card remains the same, and the PLIST refers to physical property identification numbers.

3.2.3.14 Design Constraints

Design constraint information was extracted for the Linear Statics and Normal Modes disciplines. The constraints extracted were for displacements (DCONDSP), stresses (DCONVM) and natural frequency (DCONFRQ). Stress constraints were extracted for elements (DCONVM), physical property (DCONVMP) and material identifications (DCONVMM). Design constraint information was extracted using the Open Link and Open User Interface. Constraints on displacements, frequency and stress (defined on elements) were extracted using the Open Link and constraints on stress defined on physical property and material were extracted using the Open User Interface.

Figure 25 shows a wing whose skin is made of MATERIAL-1 and the substructure is made of MATERIAL-2 (spars) and MATERIAL-3 (ribs). A displacement constraint can be defined by picking the nodes as shown in Figure 25, and the limit on the maximum displacement is defined along with the direction (2 inches in Y (Translation)

direction is shown in the figure). Similarly Figure 25 shows a constraint placed on the natural frequency of the structure for the first, second and the third modes. The minimum value is specified as 6 hz, 12 hz and 18 hz respectively on the structure. This information is then translated to the ASTROS format shown in Figure 25. The second field in the DCONDSP card identifies the constraint set number, the third field is a constraint identification number, the fourth field refers to the constraint type whether LOWER or UPPER bound on the displacement, the fifth field refers to the allowable displacement (upper bound is taken as +2 in and lower bound is taken as -2 in here), the sixth field is left blank, the seventh field refers to the node on which the constraint was placed, the eighth field refers to the component of the displacement and the ninth field refers to a real coefficient. The second field of the DCONFRQ card refers to the constraint set number, the third field to the mode number, the fourth to the constraint bound type (LOWER or UPPER) and the fifth field to the frequency limit.

Von Mises stress constraints were extracted in three ways. One, by defining the constraint on the element directly, two by defining the constraint on the physical property and three, by defining the constraint on the material. To define the constraint on the elements directly, all the elements are picked one after the other as shown in Figure 26 and the constraint value is defined as shown in Figure 26. In this figure, the constraint value is specified as 25000 psi for all the elements in the wing skin. In the next step, MIDAS scans for unused physical property tables. For example, constraints have not been defined on the substructure and hence all the physical property tables of the substructure are listed out as shown in Figure 26. Constraints are specified on all the elements belonging to the spars (say, 30000 psi). Now MIDAS further scans for unused elements and brings up MATERIAL-3 identification number and another constraint on this material (say 20000 psi) is specified. In this way, a Von

Mises stress constraint can be defined on elements, physical property and material identifications simultaneously. The ASTROS translation for these stress constraints is shown in Figure 26. The second field of the DCONVM card specifies the constraint set number, the third field the constraint limit, the fourth and the fifth are left blank, the sixth field specifies the element type and the remaining fields list all the elements. DCONVMM follows the same syntax as DCONVM until the fifth field, and its remaining fields identify the materials. DCONVMP also follows the syntax as DCONVM until the fifth field, the sixth field denotes the property type, and the remaining fields refer to the physical property identification.

3.2.3.15 Design Parameters

An option has been provided in MIDAS for the user to override default design parameters specified by ASTROS. Some of these default parameters include tolerances on constraints, objective function, design variables, etc. Since I-DEAS does not provide this facility a user-interface was developed using the Open User Interface shown in Figure 27. All the parameters have been listed, and an explanation is provided for each of these as shown in Figure 27. The user can select each of these parameters and specify the value. These modified values are then included with the ASTROS file as shown in Figure 27 using the MPPARM option.

3.2.3.16 Aero Model

Aero model data has been extracted for ASTROS to define the aerodynamic physical data (AERO) such as the reference density (of air) and reference length along with the definition of the aerodynamic macroelement (CAERO1, AEFACT) and the surface spline (SPLINE1) to relate the aerodynamic model to the structural model. Current work has focused on unsteady aerodynamics and future work will include

steady aerodynamics. Structures considered for aero modeling until now include only the aircraft wing.

A methodology different from the one discussed previously has been adopted for the extraction of aerodynamic data. Open Link and Open User Interface have been used in developing the ASTROS model for aerodynamics, but the manner in which the Open Link has been used here is different from the one described previously. MIDAS uses a 'mapping' technique to generate the aerodynamic model, since I-DEAS does not have an aero module. The model generated in I-DEAS using its conventional techniques is 'mapped' to an aerodynamic model in ASTROS. This methodology is described in detail here.

Aerodynamic model generation in MIDAS begins after the structural model has been generated. To begin the aero model generation the structural model is first 'put away' in the bin and the screen cleared completely of any geometry or finite element model. The aerodynamic model is generated in three stages. The first stage is to input all the parameters defining an aerodynamic model, the second stage is when the model is generated graphically in I-DEAS and the third stage is when this model is translated into the ASTROS format.

The first stage is to define parameters such as the number of chordwise and spanwise boxes in the aero model, the root and the tip chord, sweep angle, panel span, etc as shown in Figure 28. The form shown in the figure was developed using the Open User Interface through which a user can begin the aerodynamic analysis. Using this information the following procedure is followed to display the model graphically: MIDAS uses the "Master Modeler" task in I-DEAS along with Open Link to develop the aerodynamic model graphically. It uses data supplied by the user (shown in Figure 28) to develop this model. The first stage is to define the workplane. The dimension of the workplane should fit the aerodynamic model. For this a rough sketch of the

model can be made as shown in Figure 29. All the dimensions defined by the user are indicated in the figure. Using these dimensions the X and Y coordinates of the point A are found which determines the dimensions of the workplane. To accommodate the aero model comfortably within the workplane, the dimensions of the workplane are increased by 40 percent (arbitrarily). The next step is to use dimensions D1 to D6 and draw an outline of the aero model. The outline is drawn using the 'line' option in the 'Master Modeler' using Open Link command mnemonics. Now, using the number of divisions in the chordwise and spanwise directions, the division points in the spanwise direction and chordwise direction are computed using a sine and cosine distribution respectively. To determine the distributions the following equations are used:

Let N_s be the number of spanwise boxes, N_c be the number of chordwise boxes and $i(1,2,...y)$ be the ith division. Then the dimension 'D_i' can be determined using:

$$D_i = D4 \ Sin\frac{90i}{N_s} \qquad \text{(Sine distribution spanwise)}$$

$$D_i = D2 \ \frac{(1-\frac{Cos180i}{N_c})}{2} \qquad \text{(Cosine distribution chordwise)}$$

Dimensions D8 to D21 are then plotted on the aerodynamic model. The aerodynamic model is shown in Figure 29. After the generation of this model, the structural model is 'got' from the bin and the two models (structural and aerodynamic) are displayed simultaneously as shown in Figure 30. The significance of this superimposition is as follows: The user can verify the aero model with respect to the structural model and make changes if necessary, and in the post-processing stage for aerodynamics, the user can visualize the results relative to one another.

The aero model can be modified by the user if necessary. The user can use the I-DEAS dimension modifier directly on dimensions D1 to D21. It is to be noted here that the number of boxes in either direction cannot be modified this way. To modify the number of boxes, the user has to restart the entire process and redefine the numbers. To modify the dimensions, D8 to D11 have to be 'matched' in I-DEAS with

D2 so that if D2 is modified, the dimensions D8 to D11 change relatively, keeping up with the distribution. Similarly D12 to D15 is 'matched' with D3 and D16 to D21 is matched with D4. If the user wishes to keep the change in any dimension independent, then that particular dimension has be declared 'constant' so that its change is independent of others.

The next step in the aero analysis is to relate the aerodynamic model to the structural model. This is done through a surface spline. To define this, the 'DEFINE SPLINE' option in Figure 28 is selected. This will display the aerodynamic and the structural model side by side as shown in Figure 31. The idea is to allow the user to select the aero boxes and the corresponding grid points in a convenient manner. Figure 31 shows boxes 2, 3, 6, 7, 10, 11, 14, 15, 18, 19, 22 and 23 selected from the aerodynamic model and attached to the structural grid points 2, 3, 4, 7, 8, 9, 12, 13, 14, 17, 18, 19, 22, 23, 24, 27, 28, 29, 32, 33 and 34. This data is then used to define the surface spline for interpolating out-of-plane motion.

After defining all the data, using the 'WRITE' option shown in Figure 28, all the aerodynamic cards, namely, CAERO1 defining the aerodynamic macroelement, AEFACT defining the chordwise and spanwise divisions, AERO defining aerodynamic physical data and SPLINE1 defining the surface spline are written in ASTROS format. The CAERO1 entry (shown in Figure 30) references the aerodynamic element number, the identification numbers of AEFACT entries to define spanwise and chordwise distribution points, the X, Y and Z coordinates of points 1, root chord length, the X, Y and Z coordinates of points 4 and the tip chord length (shown in Figure 30) in that order. The AEFACT entries shown in Figure 30 list the spanwise and chordwise divisions respectively following the sine and cosine distributions respectively. The AERO entry lists the reference length (42.0) and reference density (0.0032) respectively. The SPLINE1 entry (shown in Figure 31) references the element number,

aerodynamic element number, the first and last box numbers whose motions are interpolated using the spline and a reference to a set identification number defining the grid points to which they are attached, respectively. The SET1 card lists all the grid points.

3.2.3.17 Output Data

After completion of the ASTROS run, results were extracted from its database and displayed graphically in I-DEAS. ASTROS, by itself does not have plotting capabilities. I-DEAS serves as a graphic supplement to ASTROS. The results include displacements, stress, strain energy and mode shapes (for Linear Statics and Normal Modes disciplines respectively). Future work will include results for aerodynamic analysis and optimization. Element types considered for output include CQUAD4, CSHEAR, CQDMEM1 and CROD.

The following methodology was adopted for extracting results from ASTROS: Broadly speaking, the results were first read from the ASTROS binary database and were transformed into a format readable by the I-DEAS post-processor. A file translator was developed to read the ASTROS binary database, and the results were stored in I-DEAS Open Data structures.

The first step was to access the ASTROS 'CASE' relation as shown in Figure 32 which contains the case parameters for each analysis within each boundary condition as input in the solution control packet. The 'CASE' relation has many attributes that contain information on the boundary conditions, discipline, output options, etc, defined in the solution control. MIDAS checks each of these attributes to determine the case parameters, whether analysis/optimization, boundary set numbers, single point constraint set numbers, etc as shown in Figure 32. For each of the output options in each boundary set, it checks all the element relations and its corresponding

output values. In this manner, for each boundary set, all the element relations are checked and all the results extracted as shown in Figure 32. These results are now transferred to the I-DEAS Open Data structures that stores results. The I-DEAS post-processor then uses these data structures to display the output as shown in Figure 32.

Extracting all the results simultaneously offers a distinct advantage to the user. In this way, the user is not queried all the time to input a particular option. Instead, all the results are available simultaneously, as shown in Figure 32. The list of results shown actually belongs to the I-DEAS post-processing list options. Hence the user can make full use of I-DEAS facilities to graphically pick result sets and thus minimize frequent interaction with MIDAS.

3.2.4 Programming Techniques in MIDAS

MIDAS was developed using a combination of tools. The primary tool used was the I-DEAS Open Architecture discussed in the previous chapters. Although the Open Architecture provides complete access to the I-DEAS model file, it has to be used in combination with a programming language. The C language was chosen in developing MIDAS, in combination with the Open Link, Open Data and Open User-Interface tools. All the program segments developed for MIDAS have to be run within the I-DEAS environment. This is due to the fact that each of these segments invoke the I-DEAS Open Architecture servers to provide the link to the I-DEAS database.

3.2.4.1 MIDAS Development using Open Link

Open Link has been used in a client mode in MIDAS. Open Link, as a client, queries I-DEAS for information, which acts as a server and sends results back to Open Link. Open Link provides a set of client functions, all beginning with 'crcl'. The first

function used in MIDAS is the 'crclConnect' which connects to the I-DEAS "mda" server. Once this connection is established, C procedures are invoked to extract data from the I-DEAS database along with an Open Link function. The function used to query I-DEAS database is 'crclCommandWait'. Through the 'crclOutput' command, the I-DEAS server returns a character buffer which contains all the information necessary. This character buffer is then scanned to obtain the required information for ASTROS input. For example, using the command 'crclCommandWait(server, "SE ST C #DUMP")', Open Link is querying the I-DEAS server for information about a constraint set (C). I-DEAS returns the character buffer 'result', containing this information. Using the 'sscanf' string search, the constraint set information is extracted. The string between the double quotes is the command passed to the I-DEAS server, and each component of the string refers to an I-DEAS command mnemonic. It is an abbreviation for an I-DEAS command. For example, SE for Set, ST for Set Type, C for Constraint set and #DUMP for dumping all the information to the buffer.

Since Open Link is not a direct data access like Open Data, it is considerably slower. Several methods have been implemented to speed up the Open Link procedures. One such method is to suppress the I-DEAS list region output, in which the I-DEAS server prints the character buffer 'result'. Suppressing this has speeded up the procedure considerably. Also the I-DEAS menu list has been switched off during the execution of the Open Link programs to speed up the program execution. Another recommendation to the MIDAS user to speed up the procedure is to iconify the I-DEAS graphic window while running the program.

3.2.4.2 MIDAS Development using Open Data

Open Data provides direct data access to the I-DEAS database. Open Data provides functions separate from Open Link which are used for data access. In segments

where Open Data is used, MIDAS first connects to the Open Data FEM service using the function 'oa_FemInit'. The C procedures are then invoked to extract data from the I-DEAS database. Using the function 'oa_QueryFemModel' the I-DEAS model ID is first queried, and with this information, the I-DEAS database is further queried for information. Each model segment is associated with a function. For example, to query elements, the function oa_QueryFemElement is used which returns a data structure containing the attributes of the elements which can be directly noted without having to scan any strings.

3.2.4.3 MIDAS Development using Open User Interface

A graphical user-interface has been provided in MIDAS at places where information for ASTROS was not available in I-DEAS. This graphic user interface was developed using X window programming techniques using the Motif toolkit. The industry standard X window system adopted in this application consists of Xlib and XtIntrinsic functions to make dialog windows, labels, push-buttons and text widgets. Functions are programmed in C in conjunction with XtIntrinsics and associated with push-buttons to perform the specific functions. For example, functions were written to execute specific segments in MIDAS, note all the information entered by the user, etc. Text widgets provided a means to enter information, and labels were used to convey information to the user.

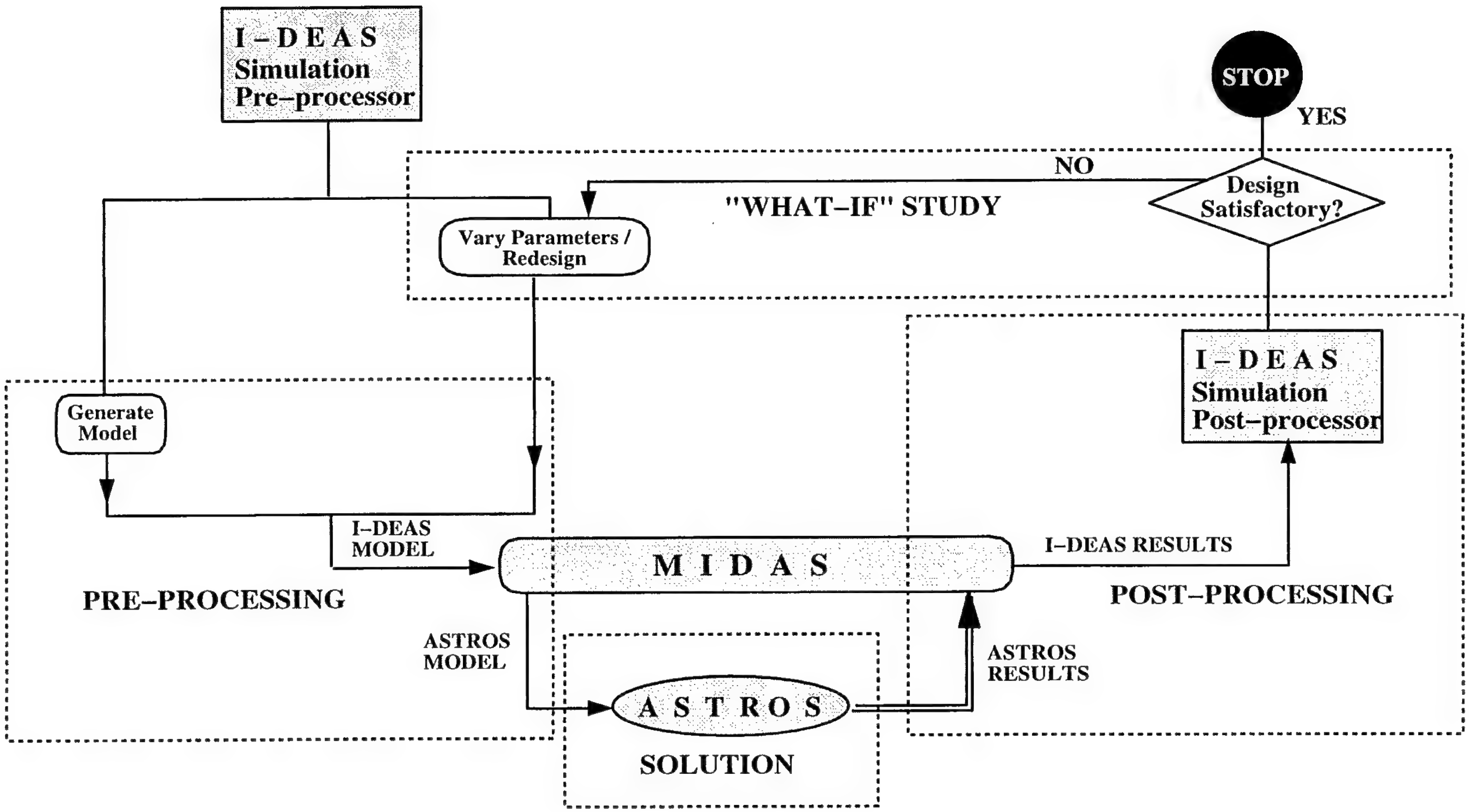


Figure 3. Design Phases in MIDAS

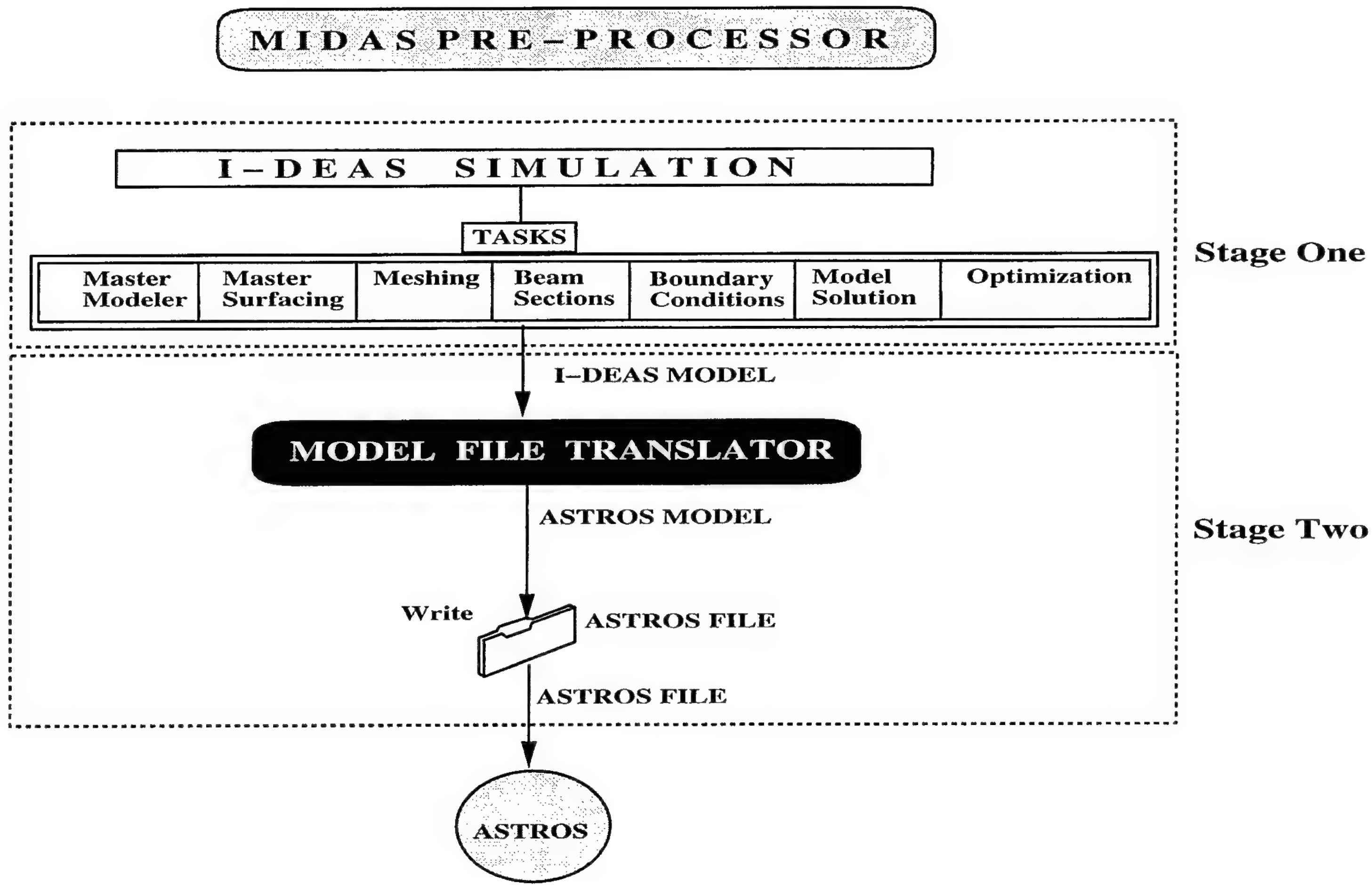


Figure 4. Pre-processing Stages in MIDAS

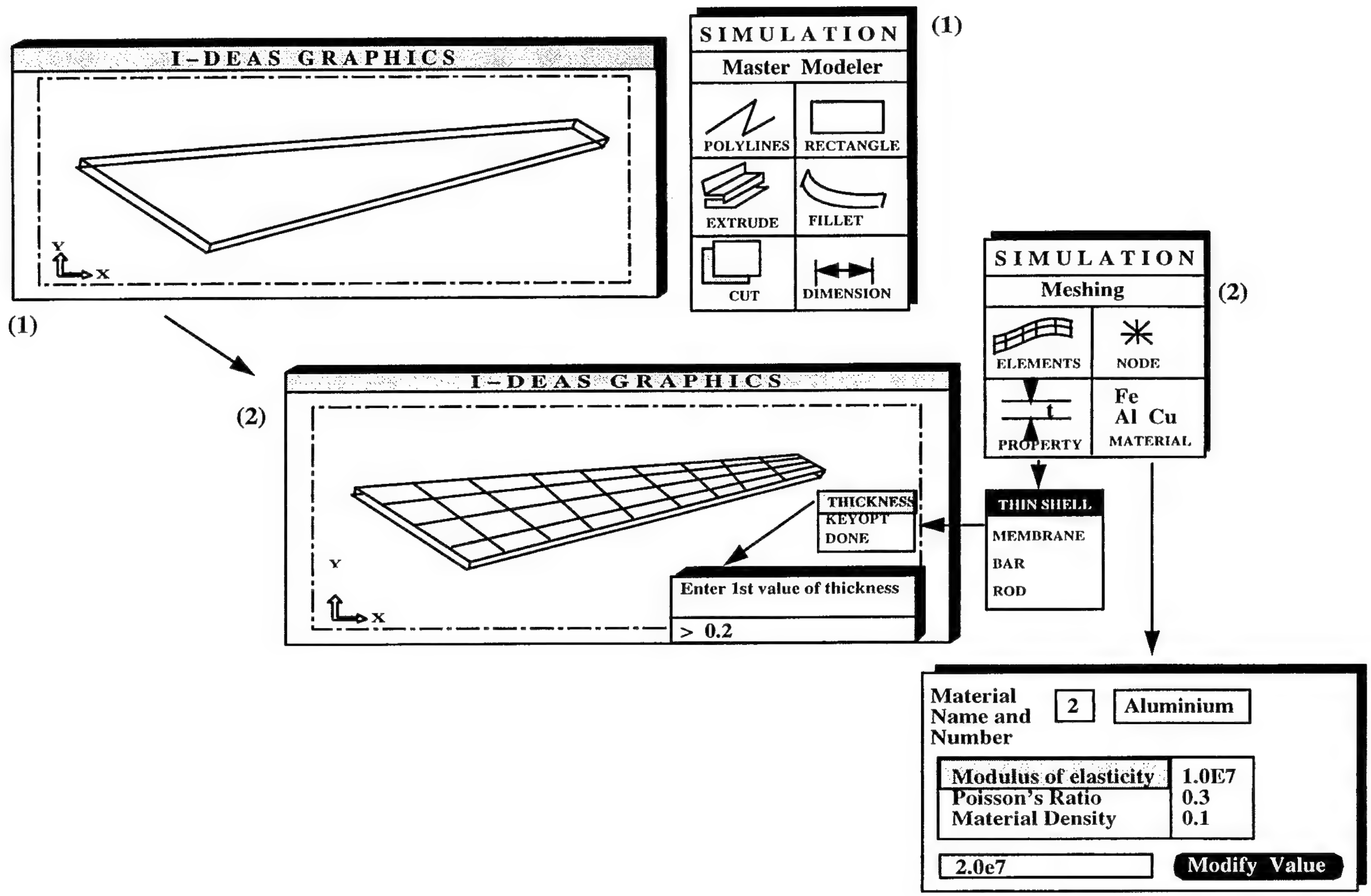


Figure 5a. Stage One in Pre-processing: Model Building in I-DEAS

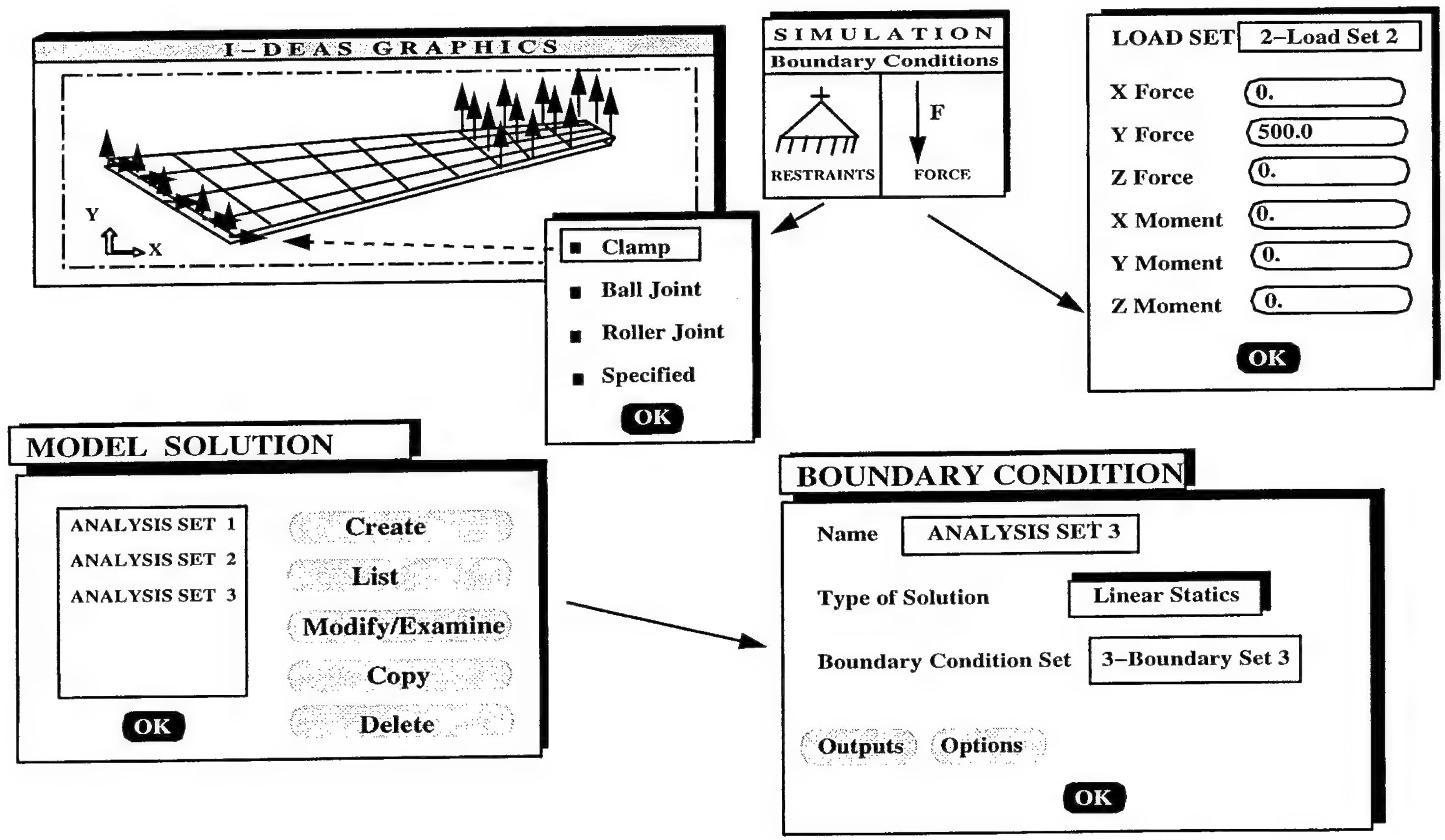


Figure 5b. Stage One in Pre-processing: Model Building in I-DEAS

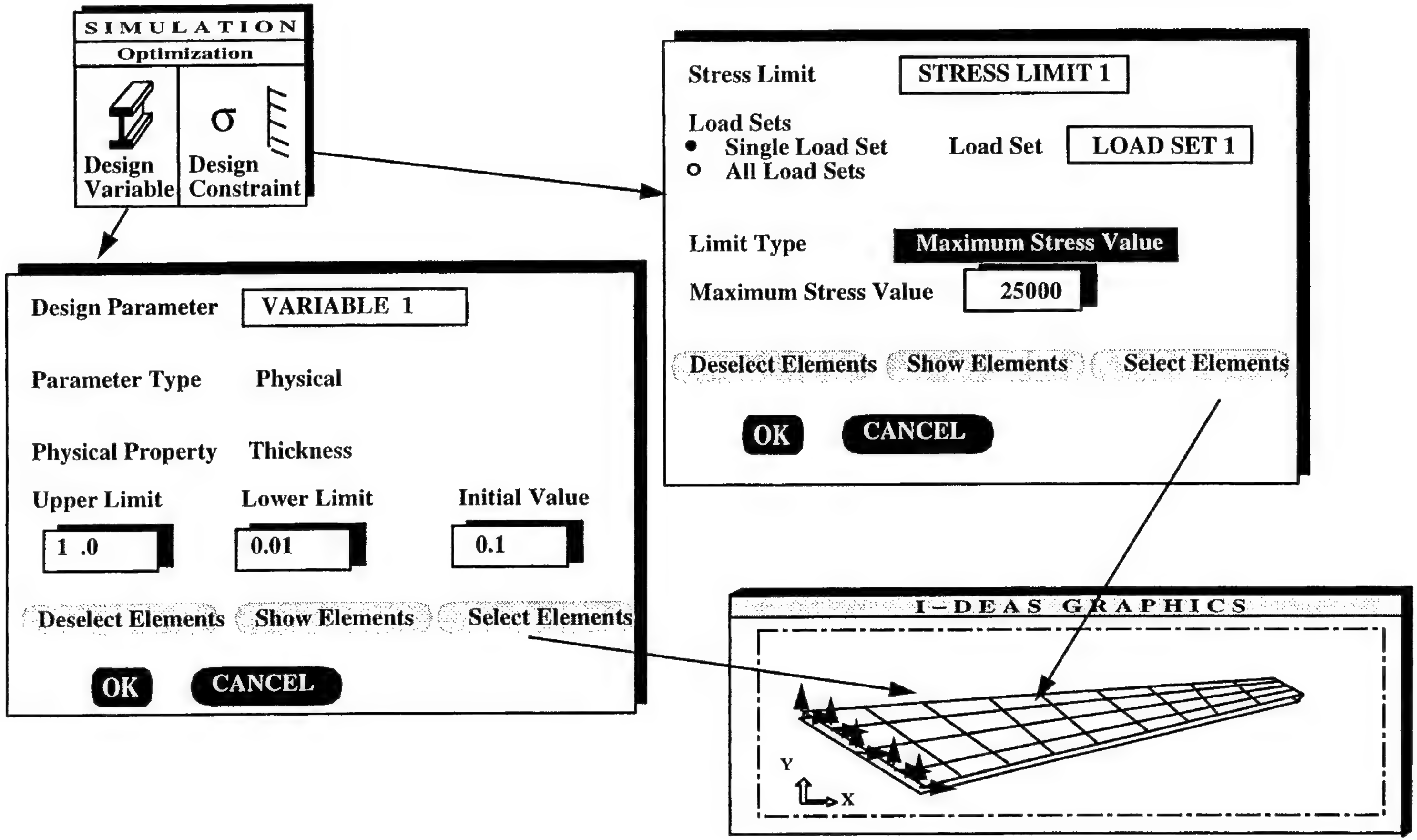


Figure 5c. Stage One in Pre–processing: Model Building in I–DEAS

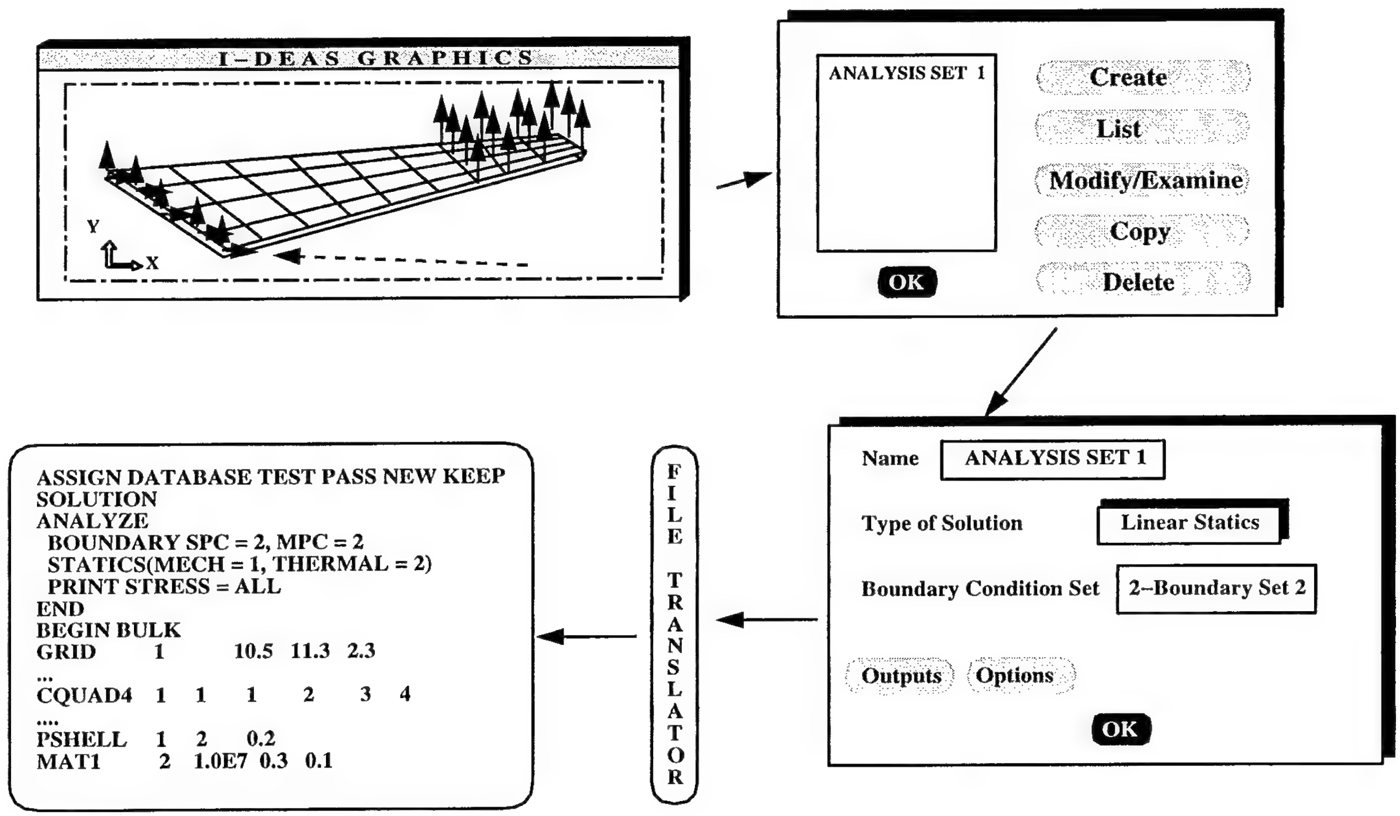


Figure 6. Stage Two in Pre-processing: Model Translation from I-DEAS to ASTROS

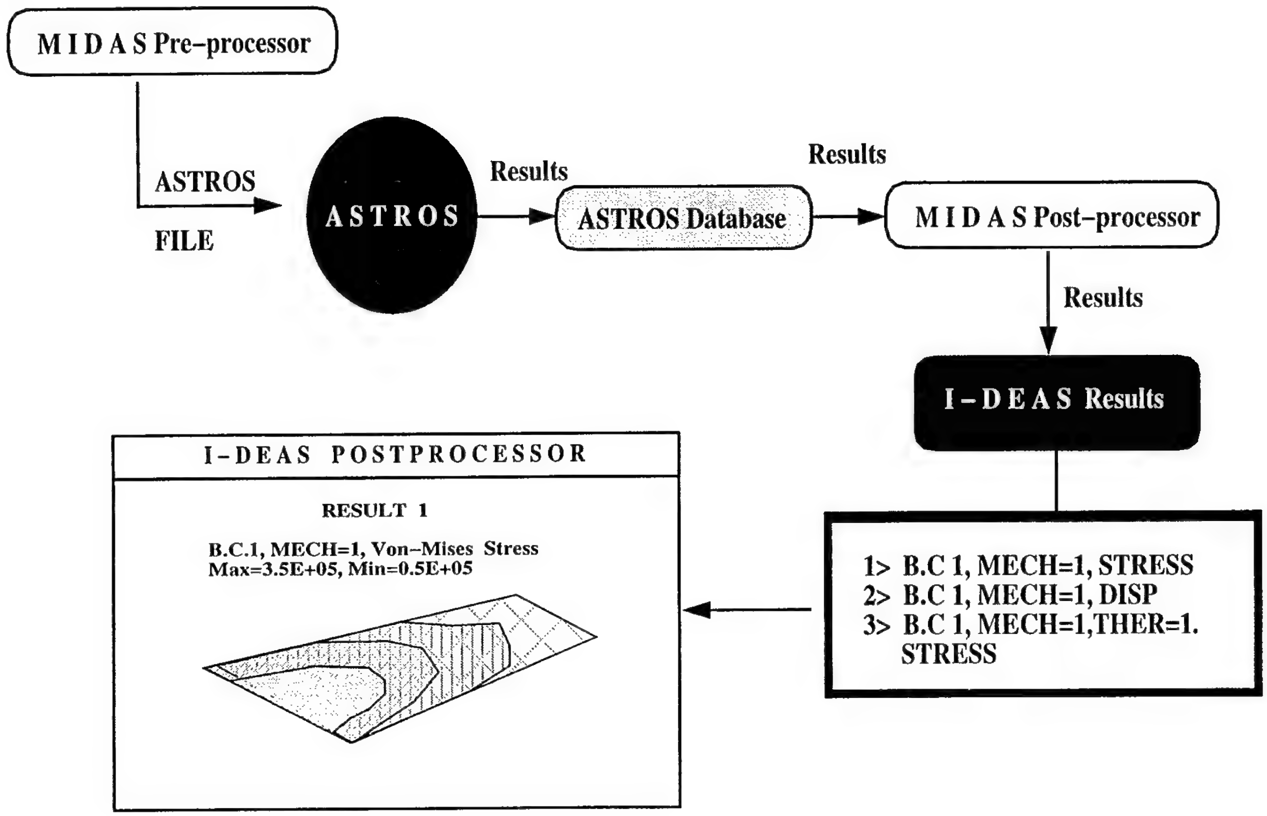


Figure 7. Post–processing in MIDAS

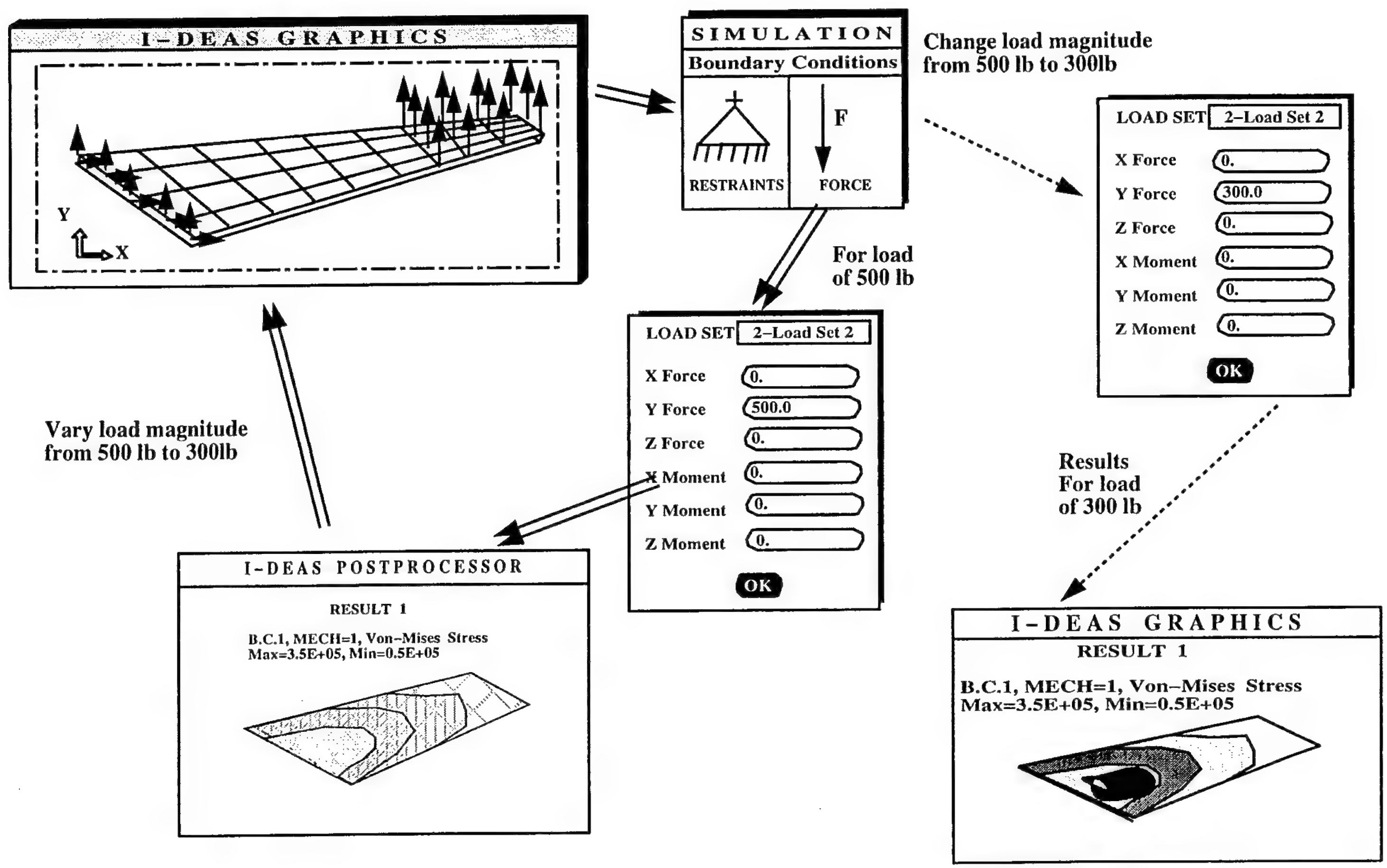


Figure 8. "What-if" studies in MIDAS

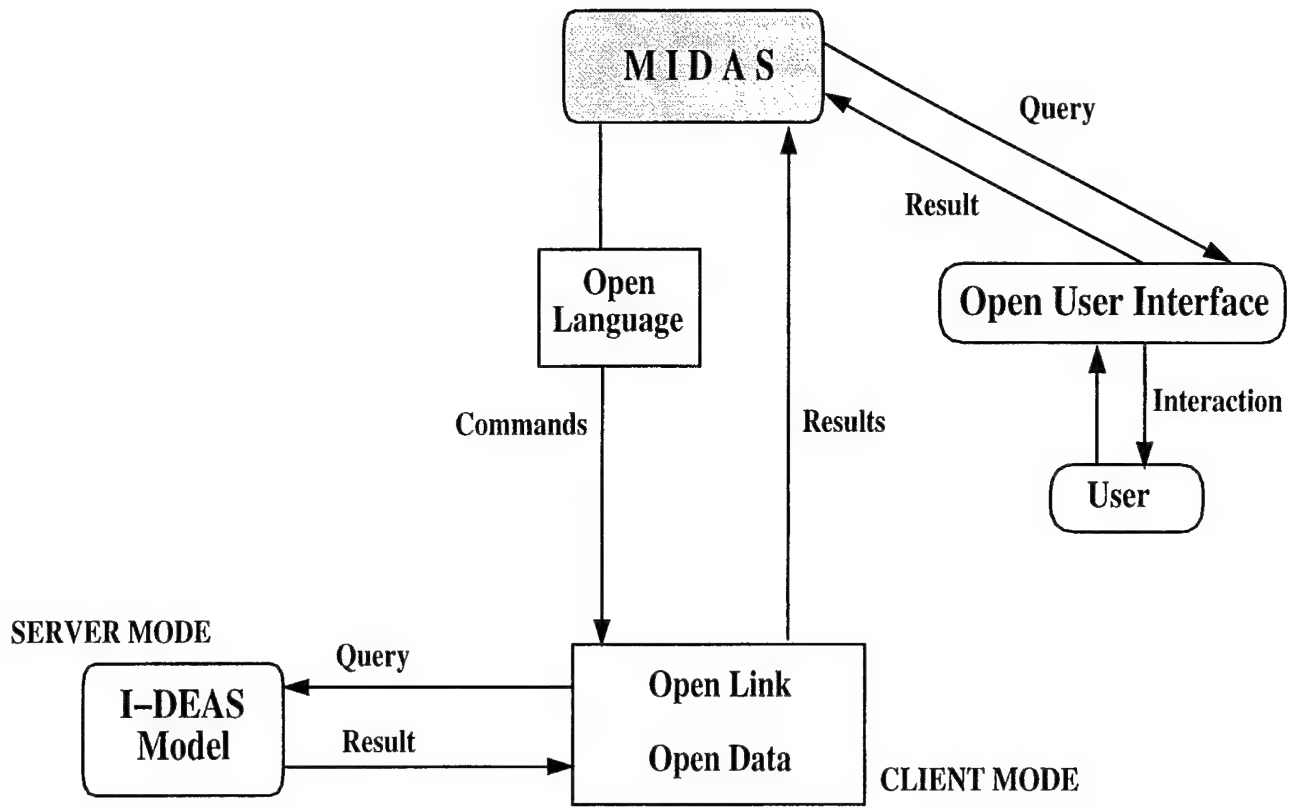


Figure 9. MIDAS Interaction with Open Architecture

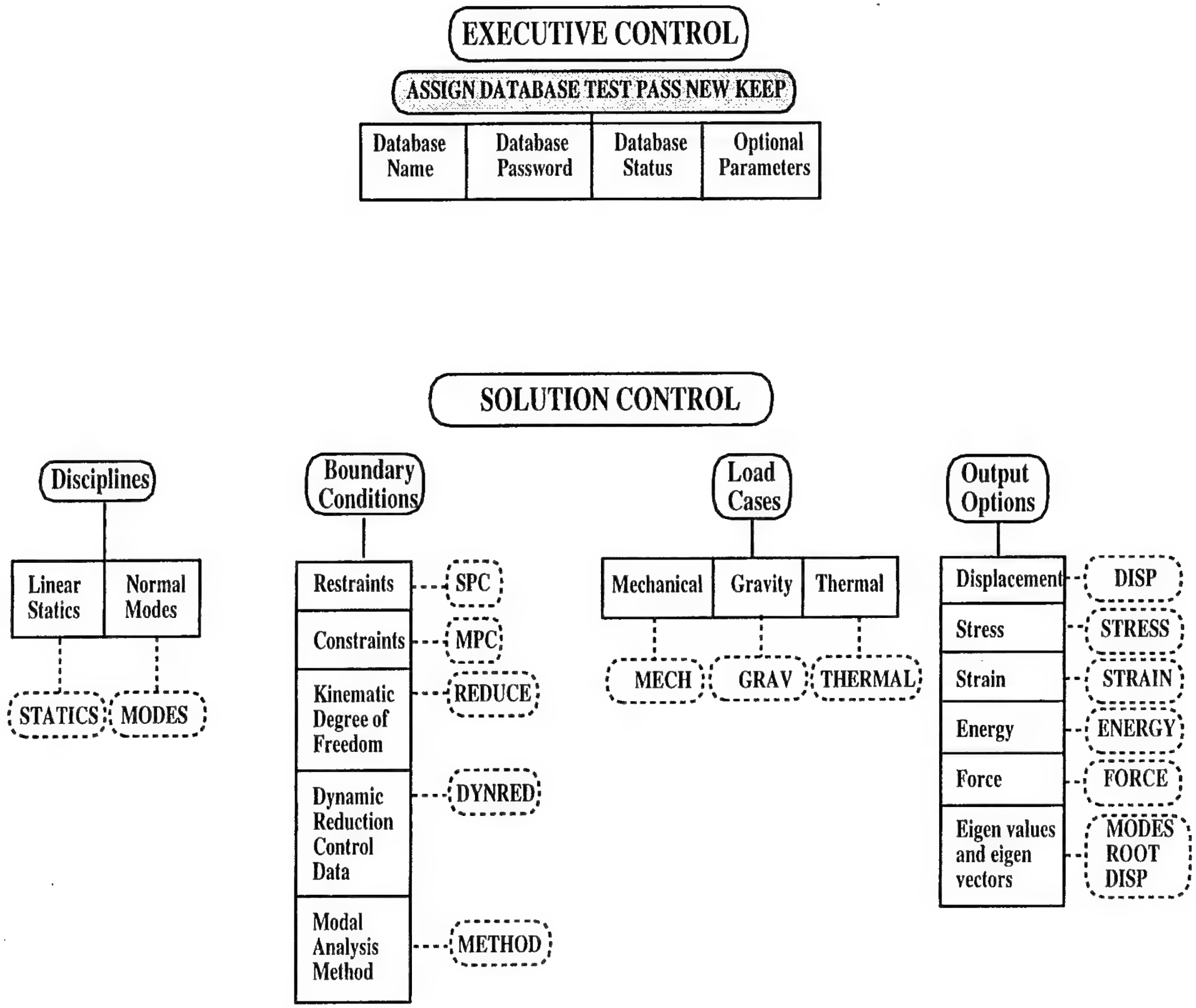


Figure 10. ASTROS Executive Control and Solution Control Data in MIDAS

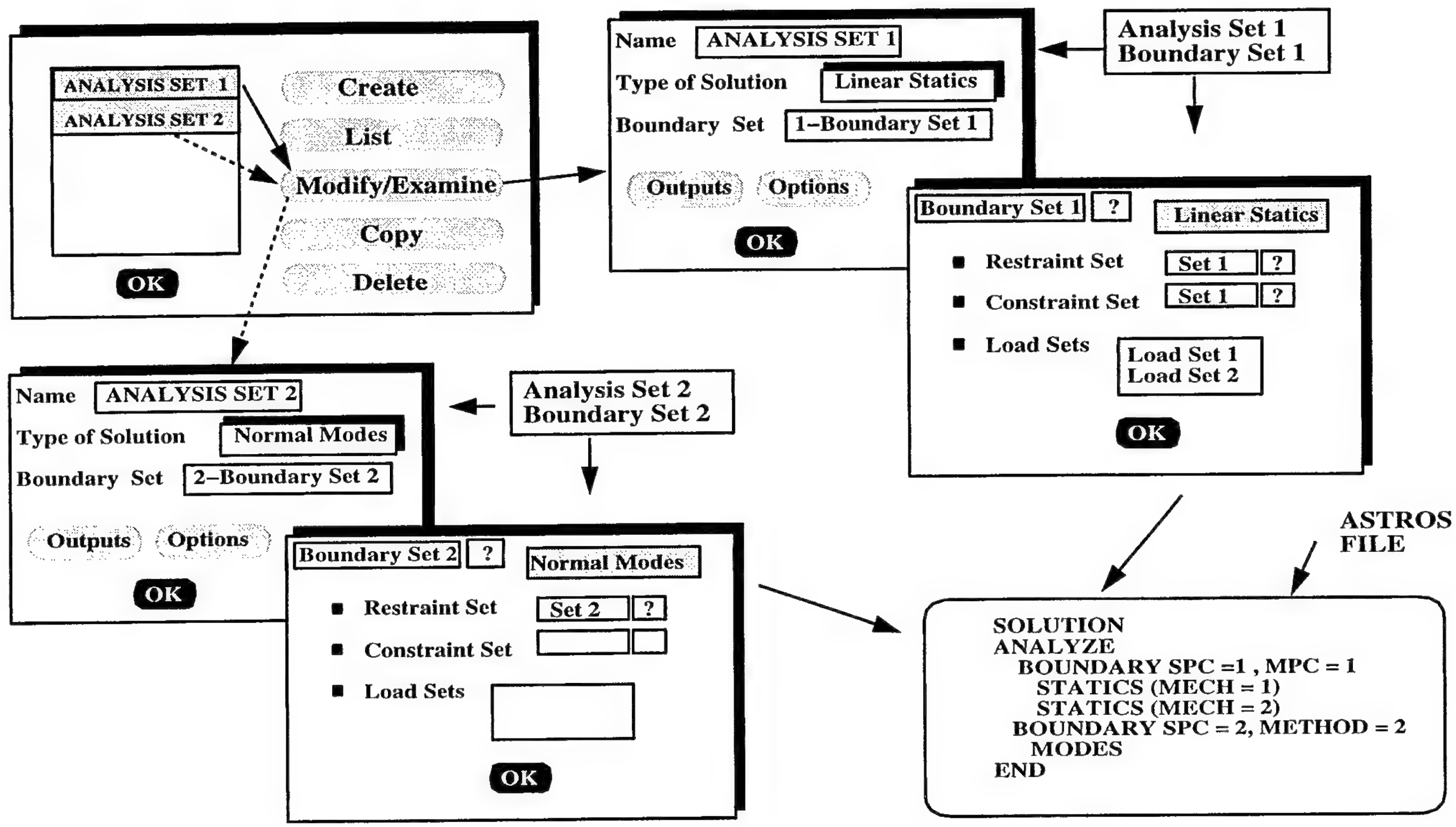


Figure 11. An Example of Multiple Boundaries and Multiple Discipline Translations in MIDAS

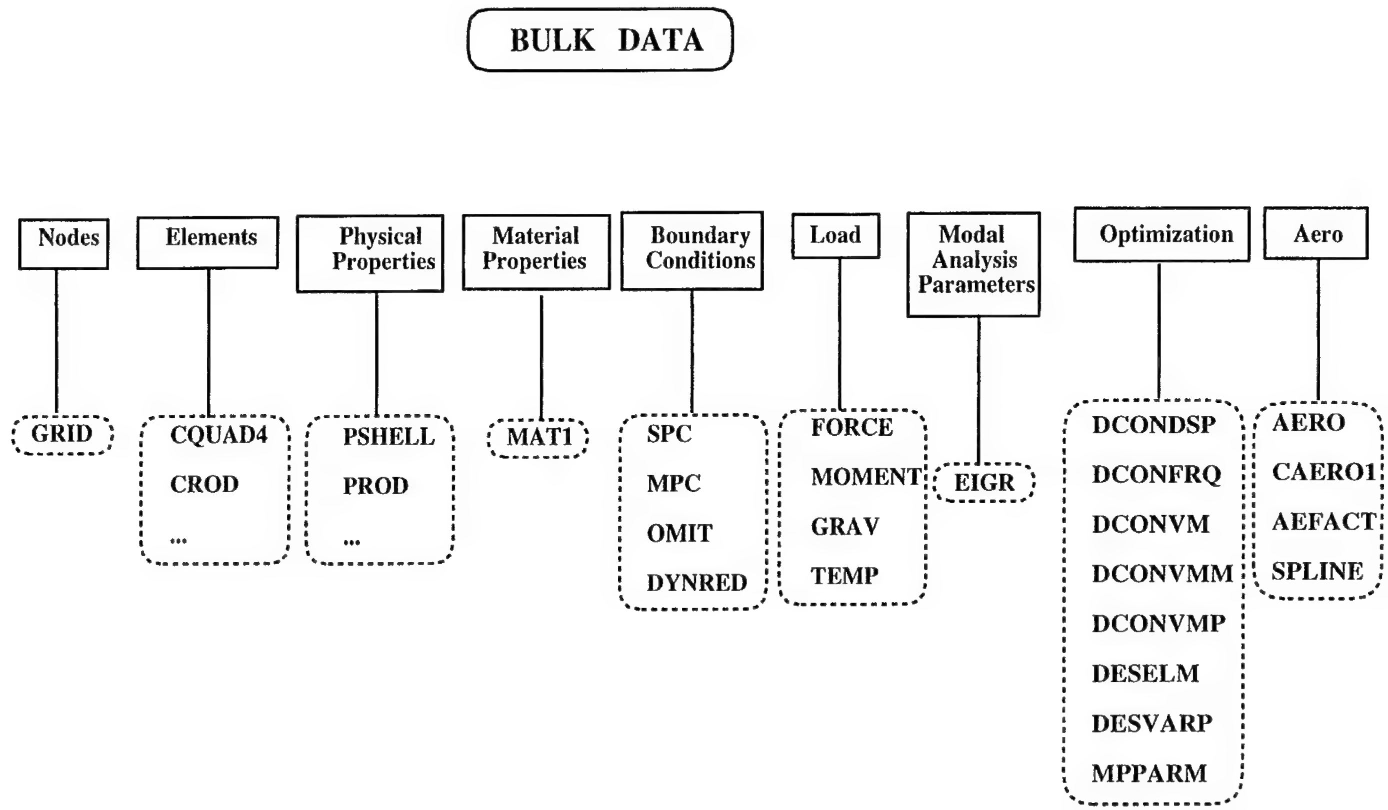


Figure 12. ASTROS Bulk Data in MIDAS

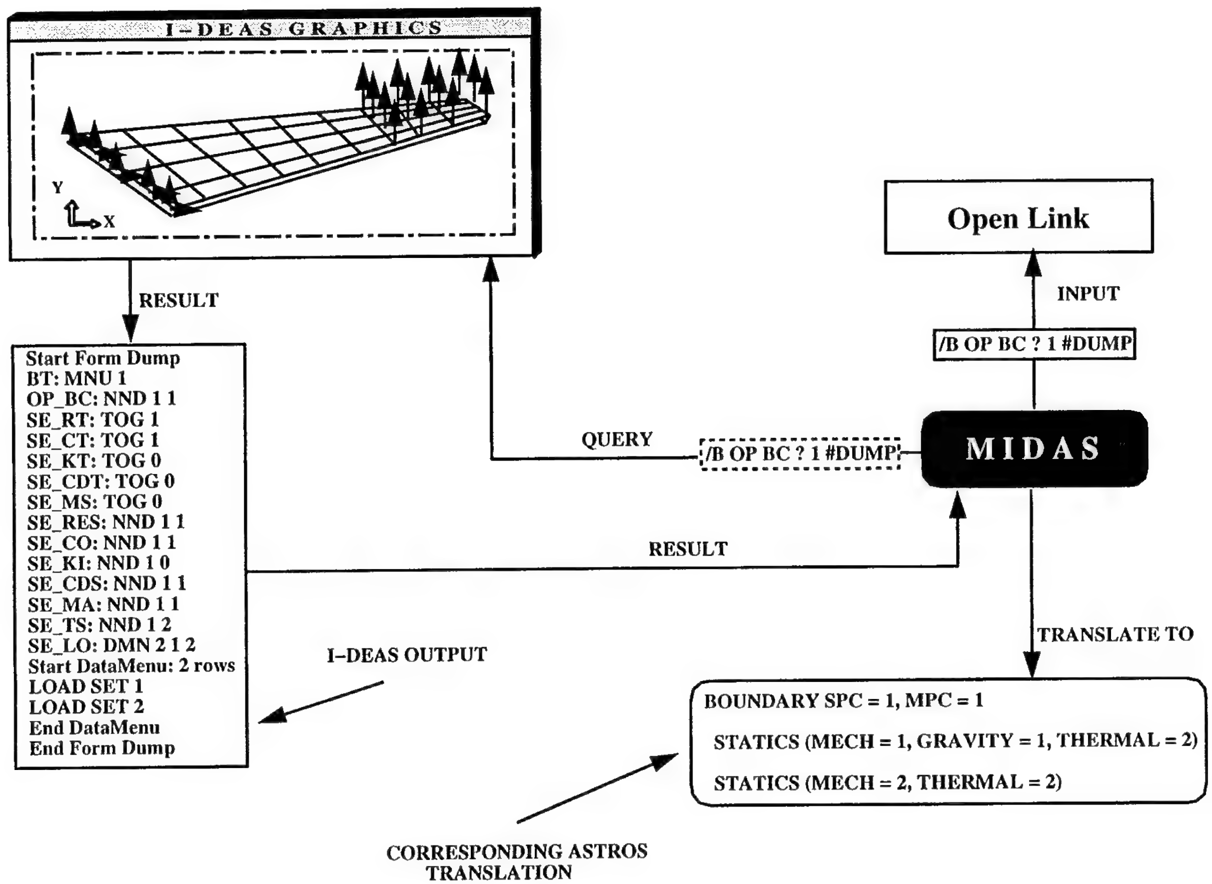


Figure 13. An Example of Model Translation Using Open Link (For Single Boundary Condition and Discipline)

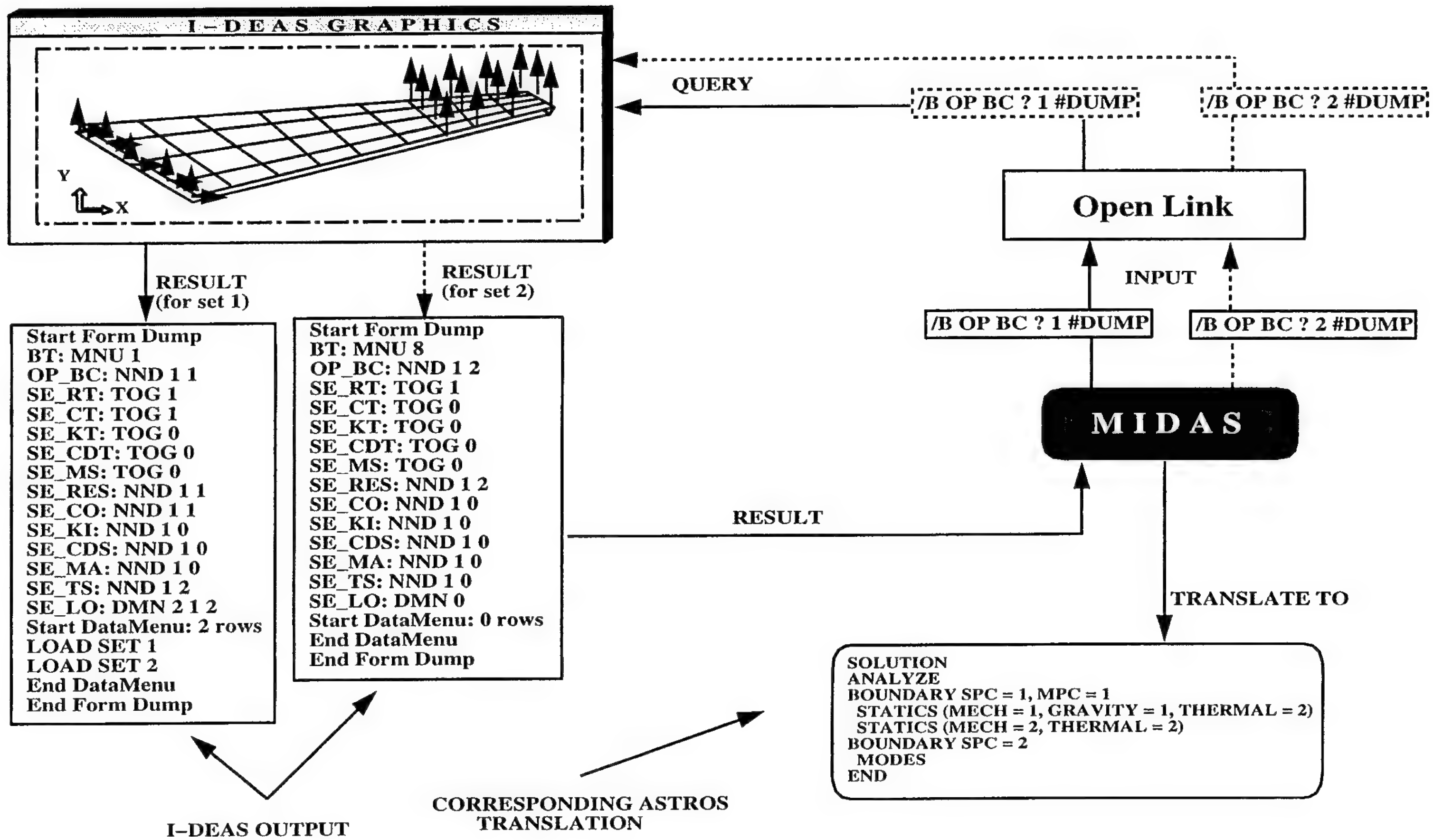


Figure 14. An Example of Model Translation Using Open Link (Multiple Boundary Conditions and Disciplines)

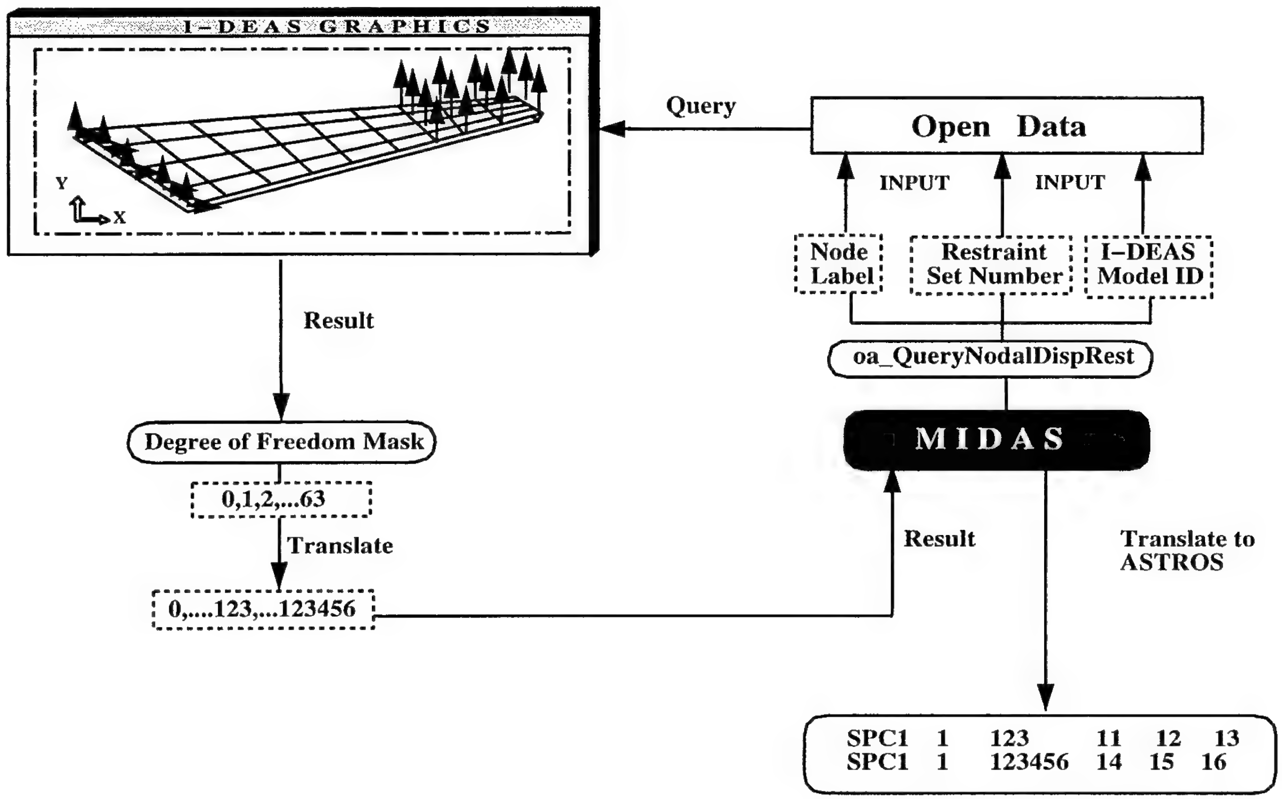


Figure 15. An Example of Model Translation Using Open Data (Single Point Constraints)

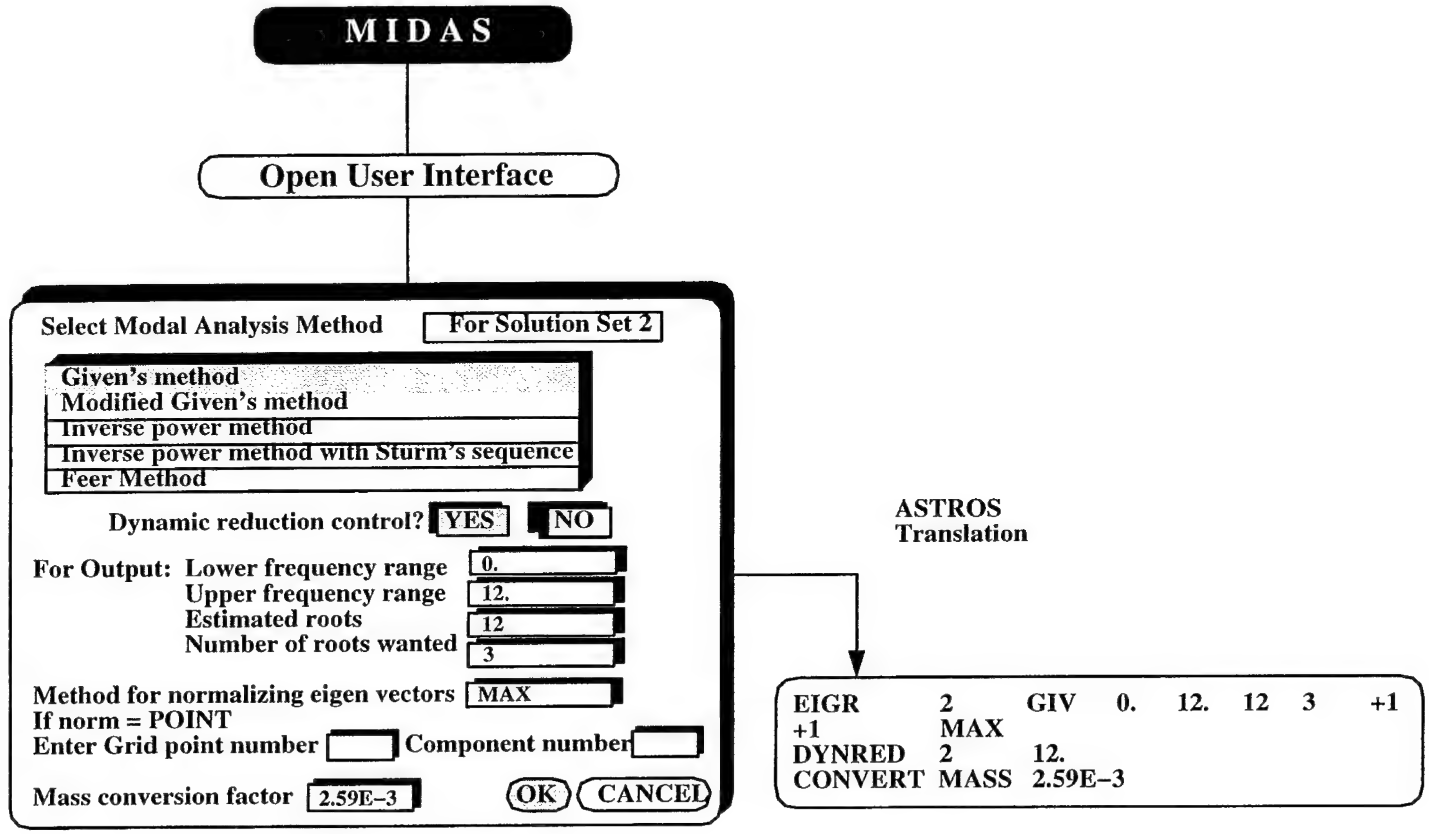


Figure 16a. An Example of Model Translation Using Open User Interface (Modal Analysis Parameters)

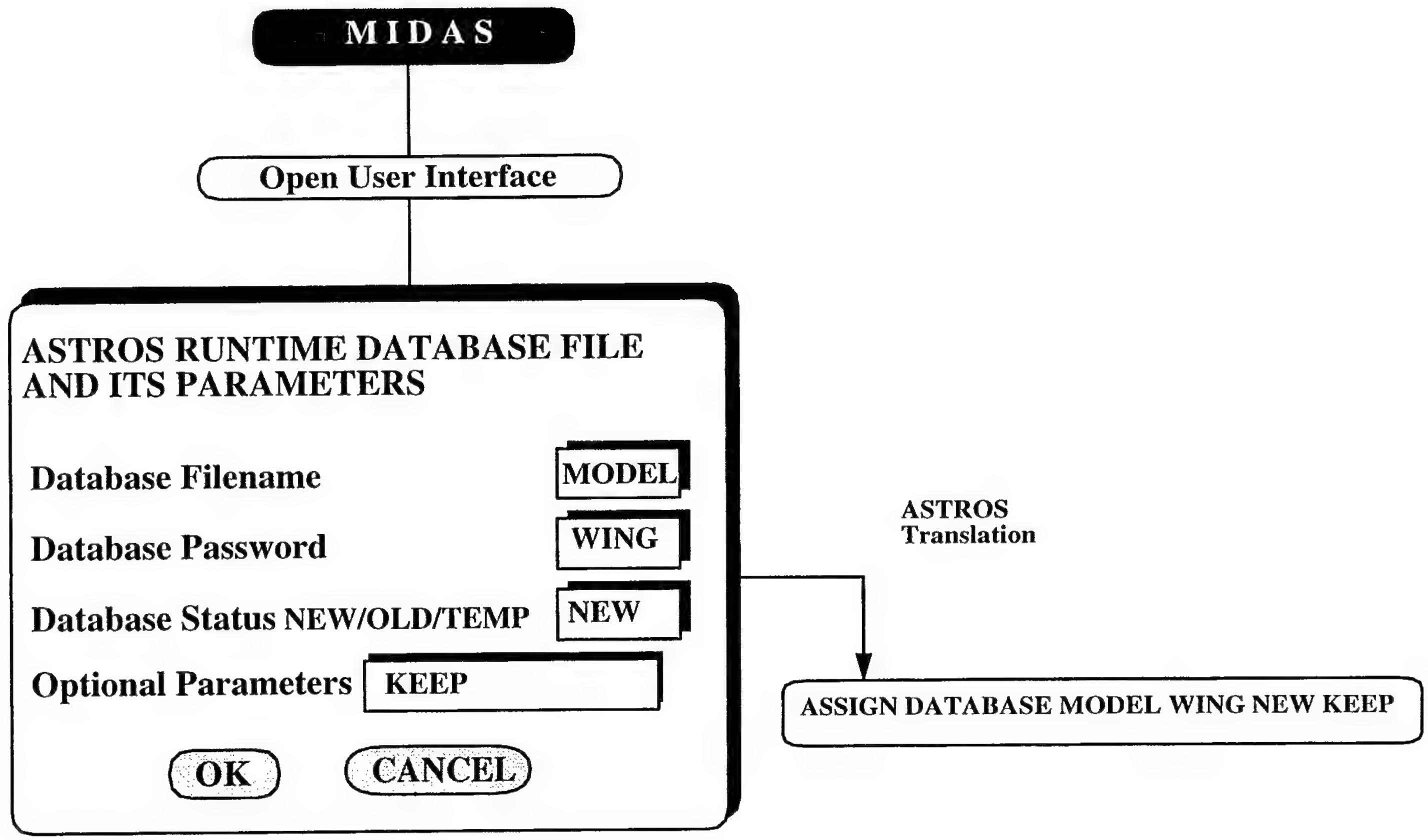


Figure 16b. An Example of Model Translation Using Open User Interface (Executive Control Parameters)

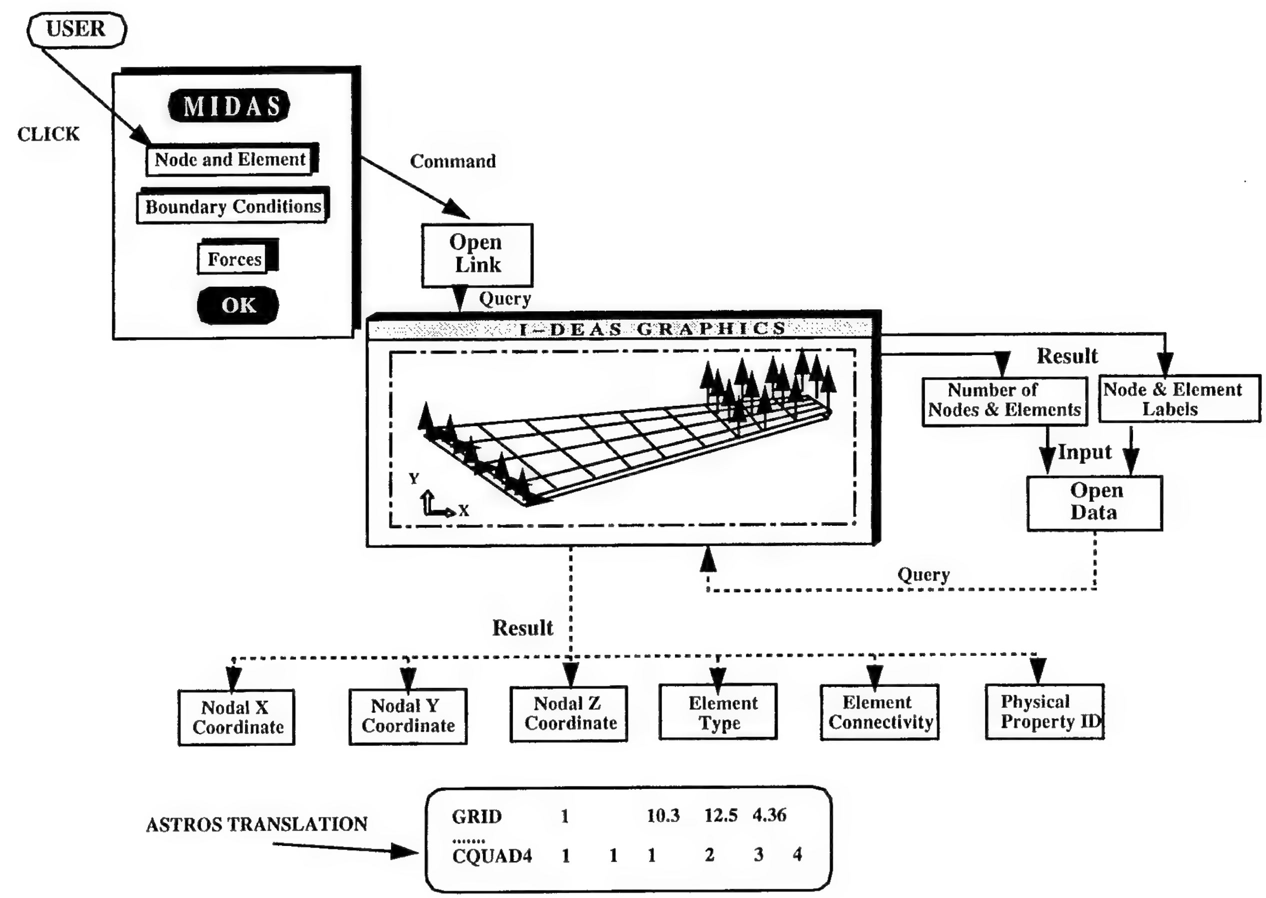


Figure 17. An Example of Node and Element Generation in MIDAS

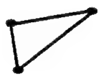
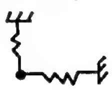
I–DEAS Element		ASTROS Element	Property Type
	Thin shell linear quadrilateral	CQUAD4	PSHELL
	Thin shell linear triangle	CTRIA3	PSHELL
	Plane stress linear quadrilateral	CSHEAR	PSHEAR
	Membrane linear quadrilateral	CQDMEM1	PQDMEM1
	Membrane linear triangle	CTRMEM	PTRMEM
	Linear beam	CBAR	PBAR
	Rod	CROD	PROD
	Linear Solid	CIHEX1	PIHEX
	Parabolic Solid	CIHEX2	PIHEX
	Cubic Solid	CIHEX3	PIHEX
	Node–node translational spring	CELAS2	
	Node–ground translational spring	CELAS2	
	Node–node rotational spring	CELAS2	
	Node–ground rotational spring	CELAS2	
	Lumped Mass	CONM2	

Figure 18. I–DEAS to ASTROS Element Mapping

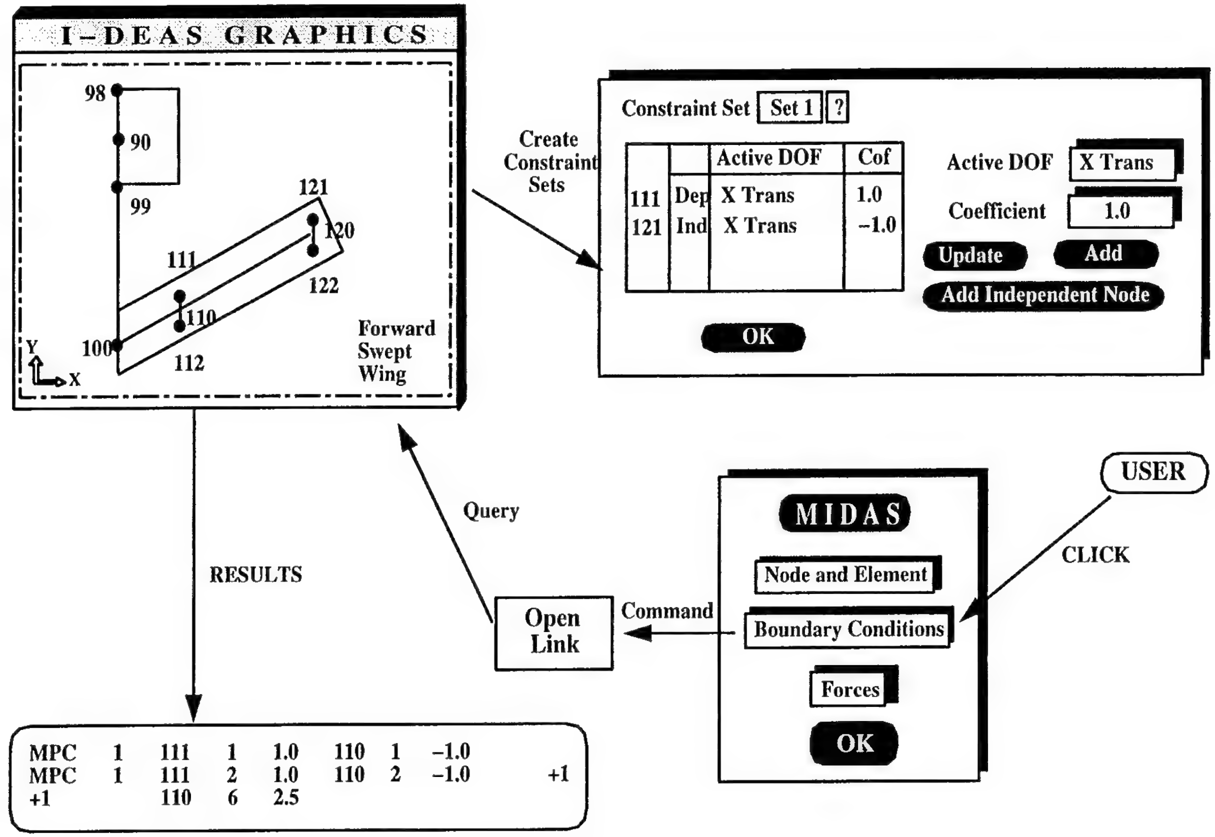


Figure 19. An Example of Boundary Condition Extraction in MIDAS (Constraint Equations)

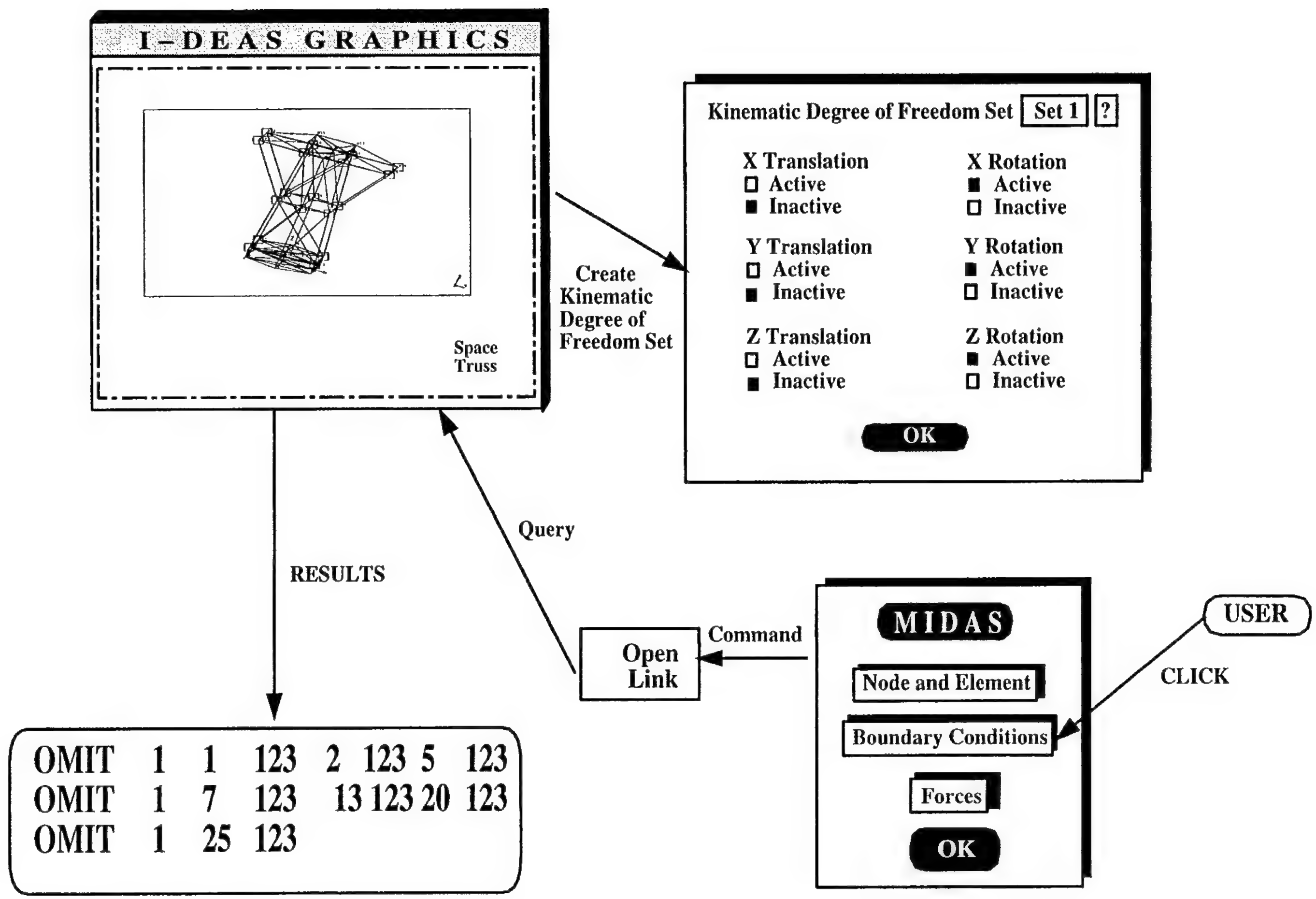


Figure 20. An Example of Boundary Condition Extraction in MIDAS (Kinematic Degree of Freedom)

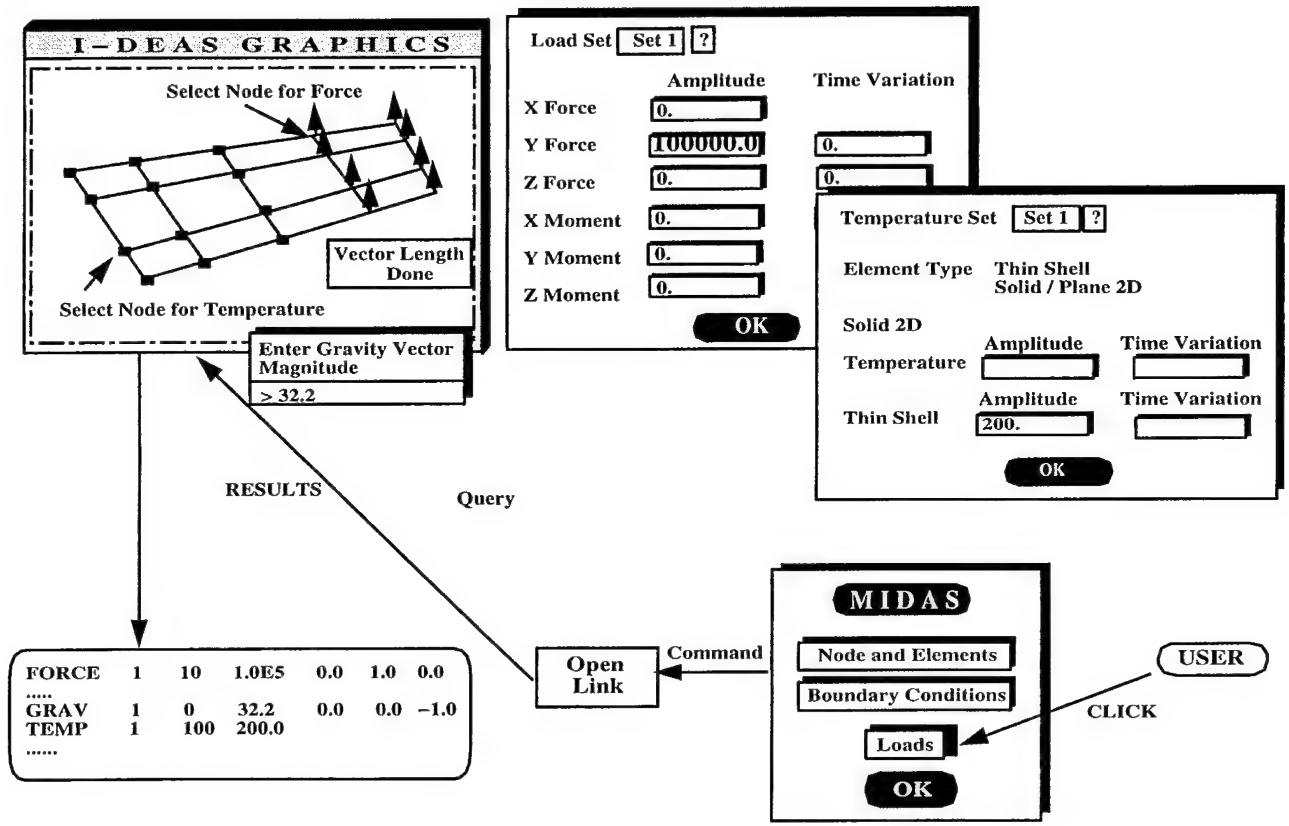


Figure 21. An Example of Load Extraction in MIDAS (Mechanical, Gravity and Thermal)

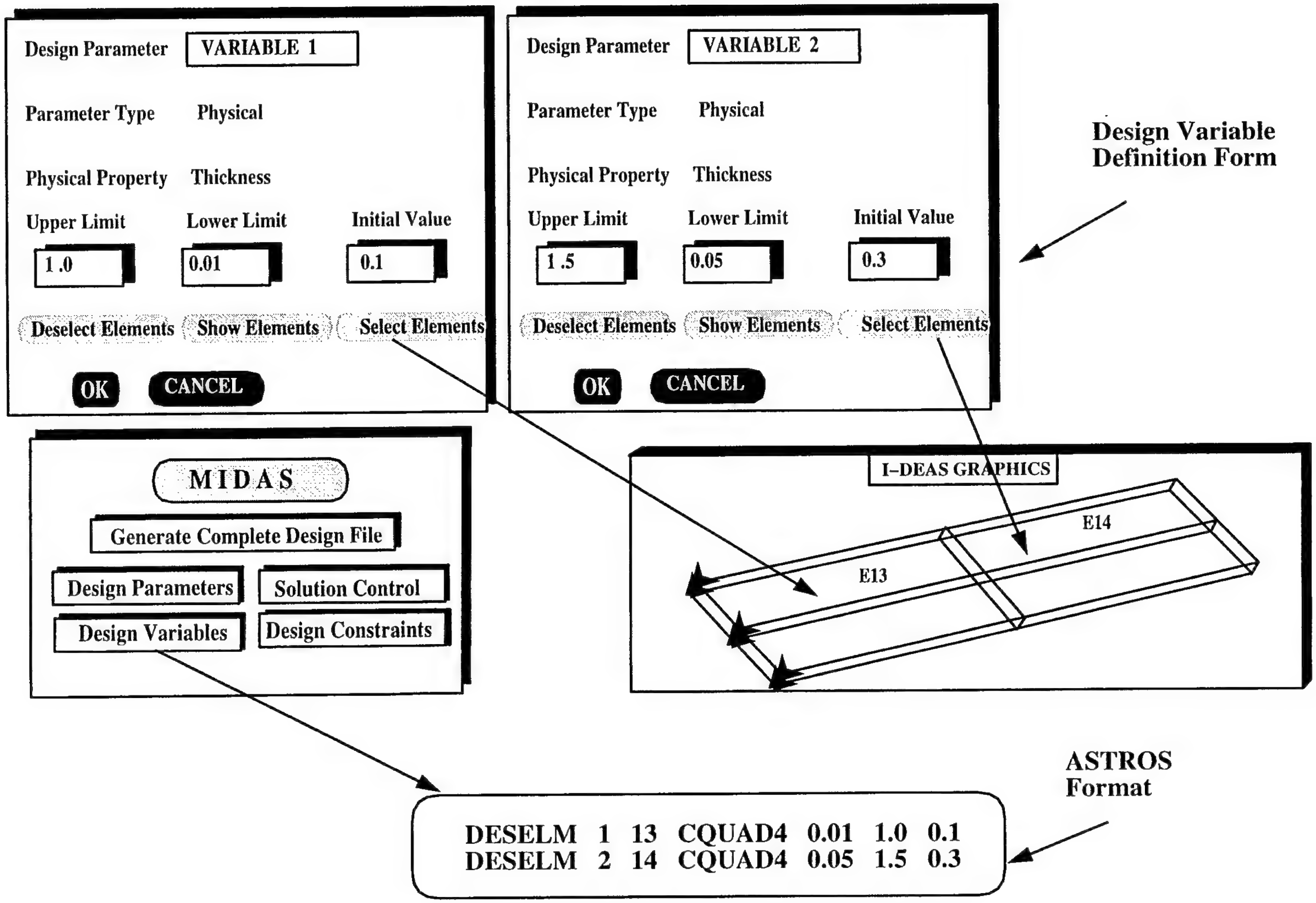


Figure 22. An Example of Design Variable Extraction in MIDAS For An Uniquely Associated Design Variable

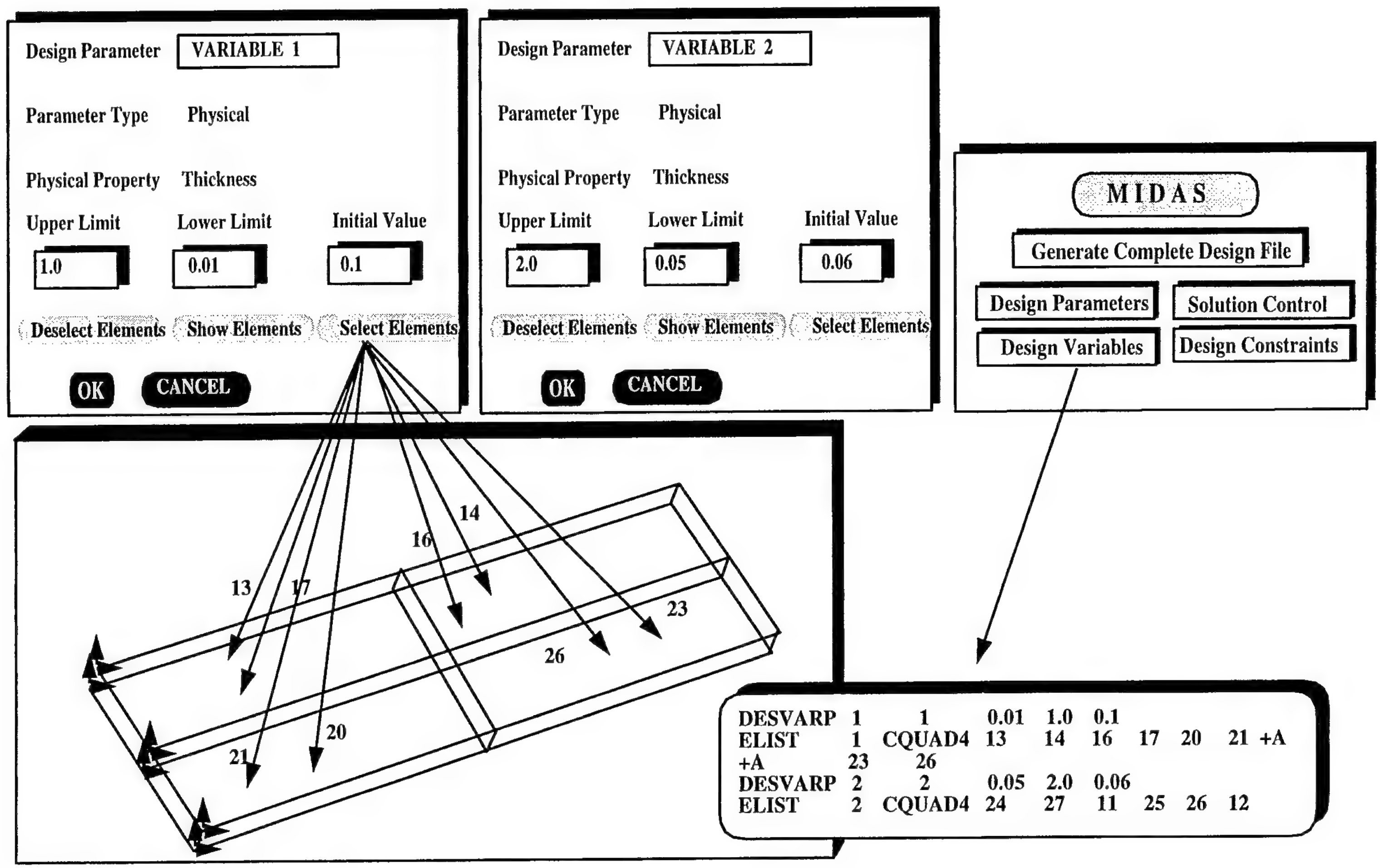


Figure 23. An Example of Design Variable Extraction in MIDAS For Direct Element Linking

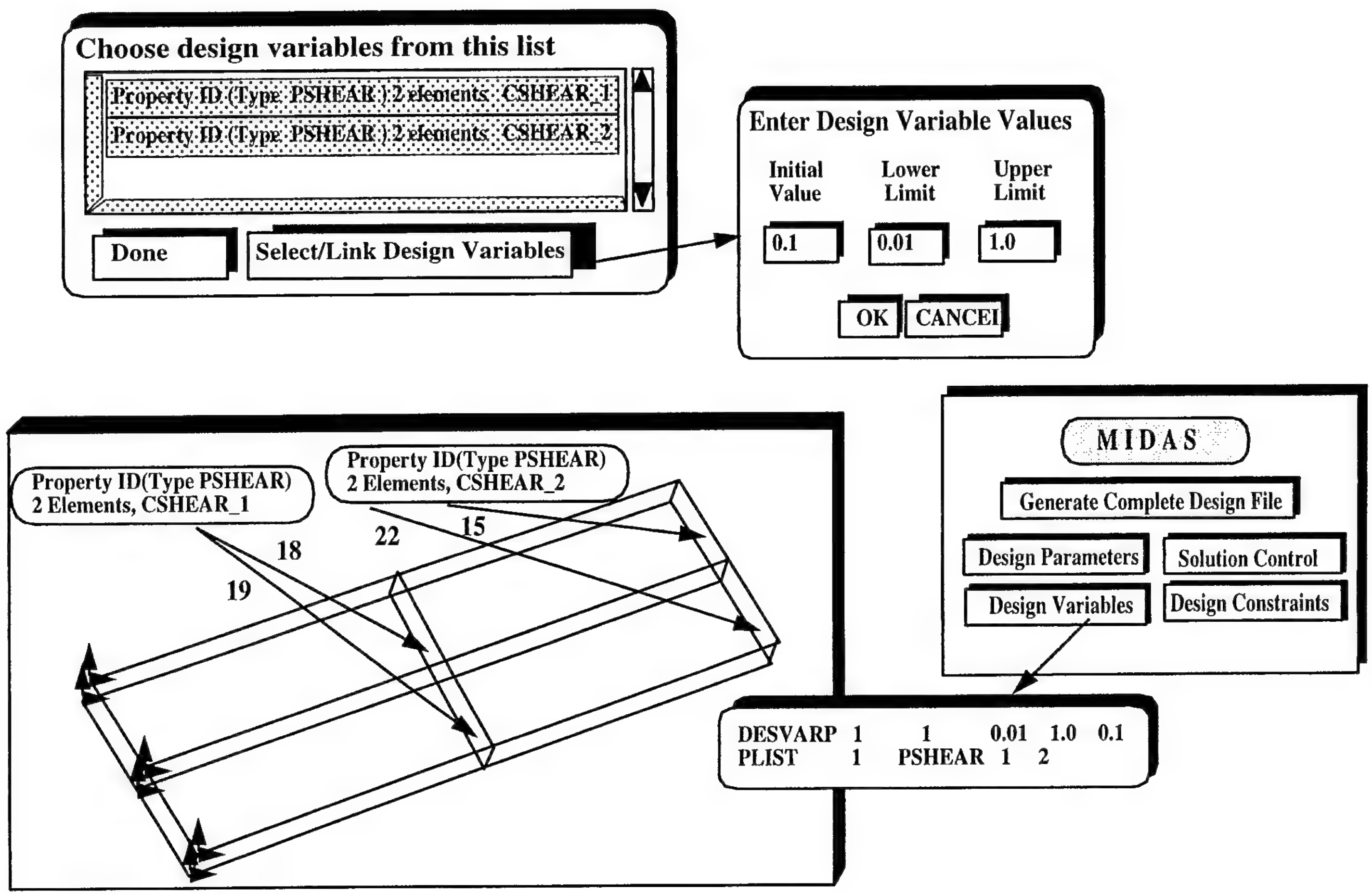


Figure 24. An Example of Design Variable Extraction in MIDAS For Element Linking Through Physical Property

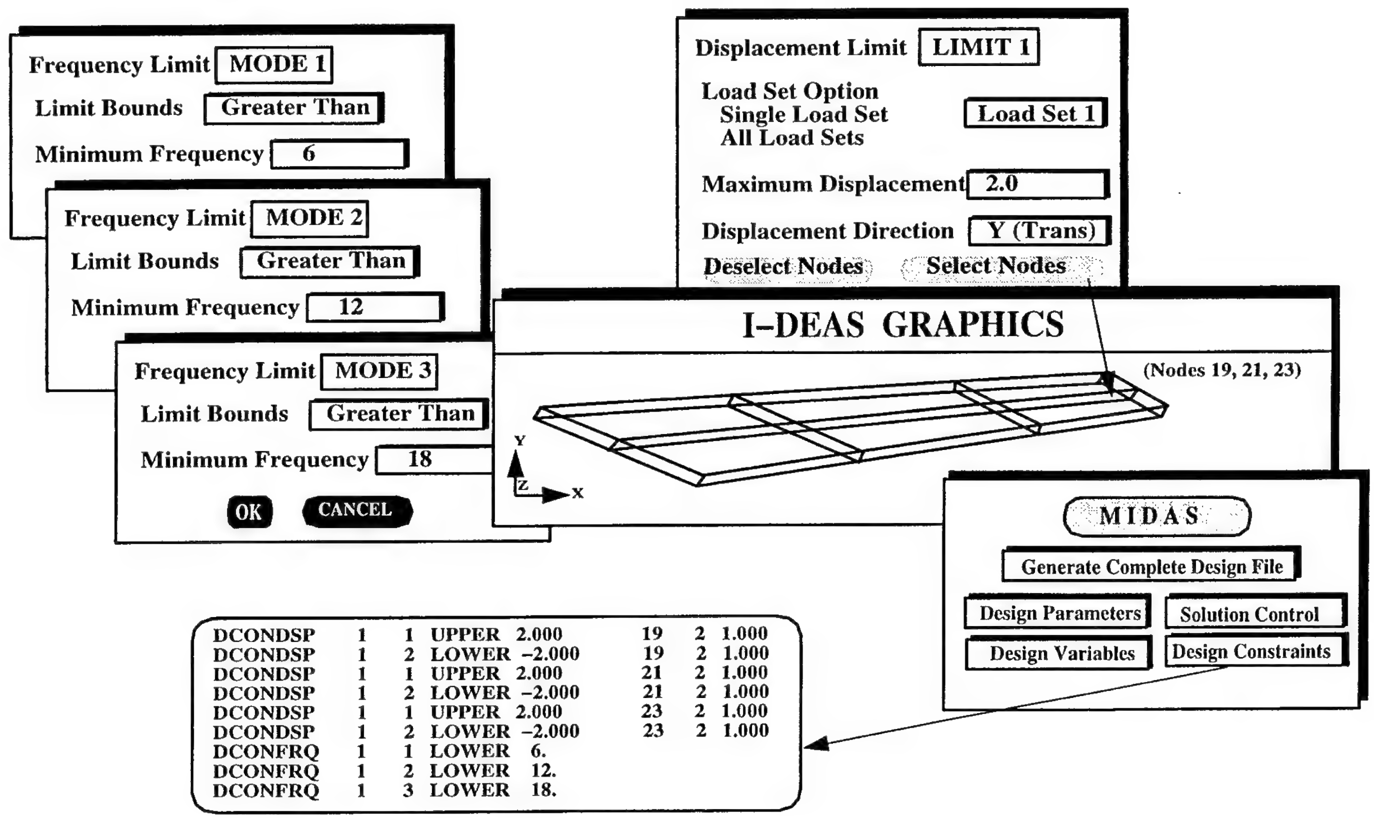


Figure 25. An Example of Design Constraint Extraction in MIDAS For Displacement and Frequency Constraints

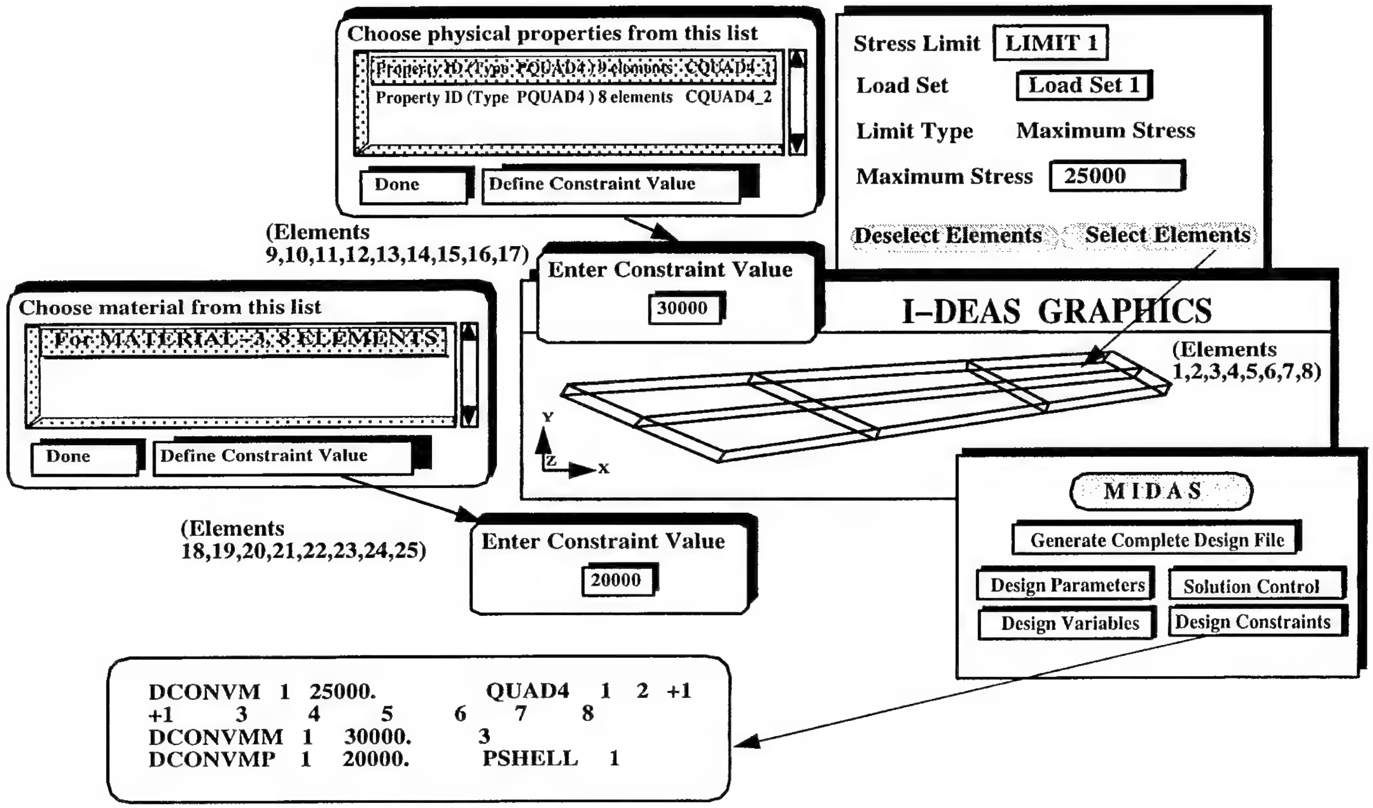


Figure 26. An Example of Design Constraint Extraction in MIDAS For Stress Constraints

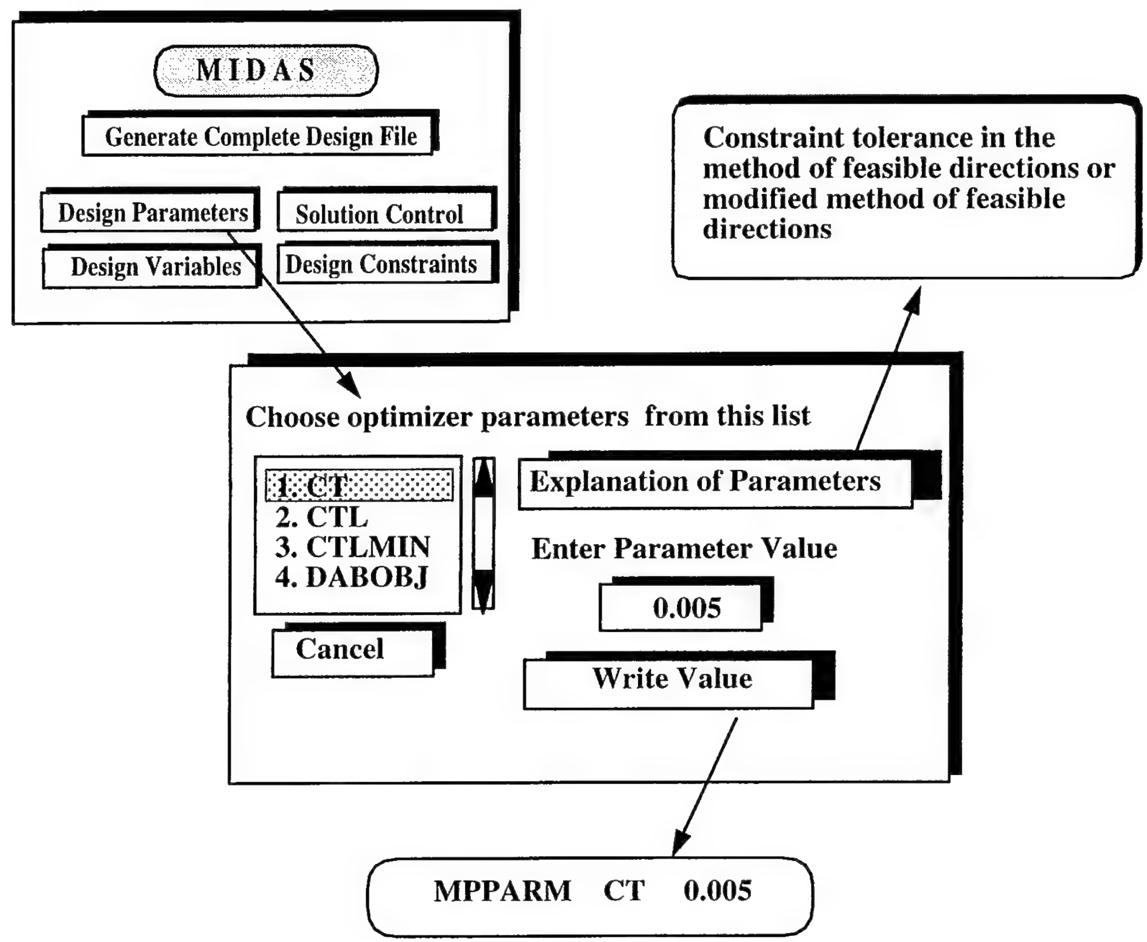


Figure 27. An Example of Design Parameter Definition in MIDAS

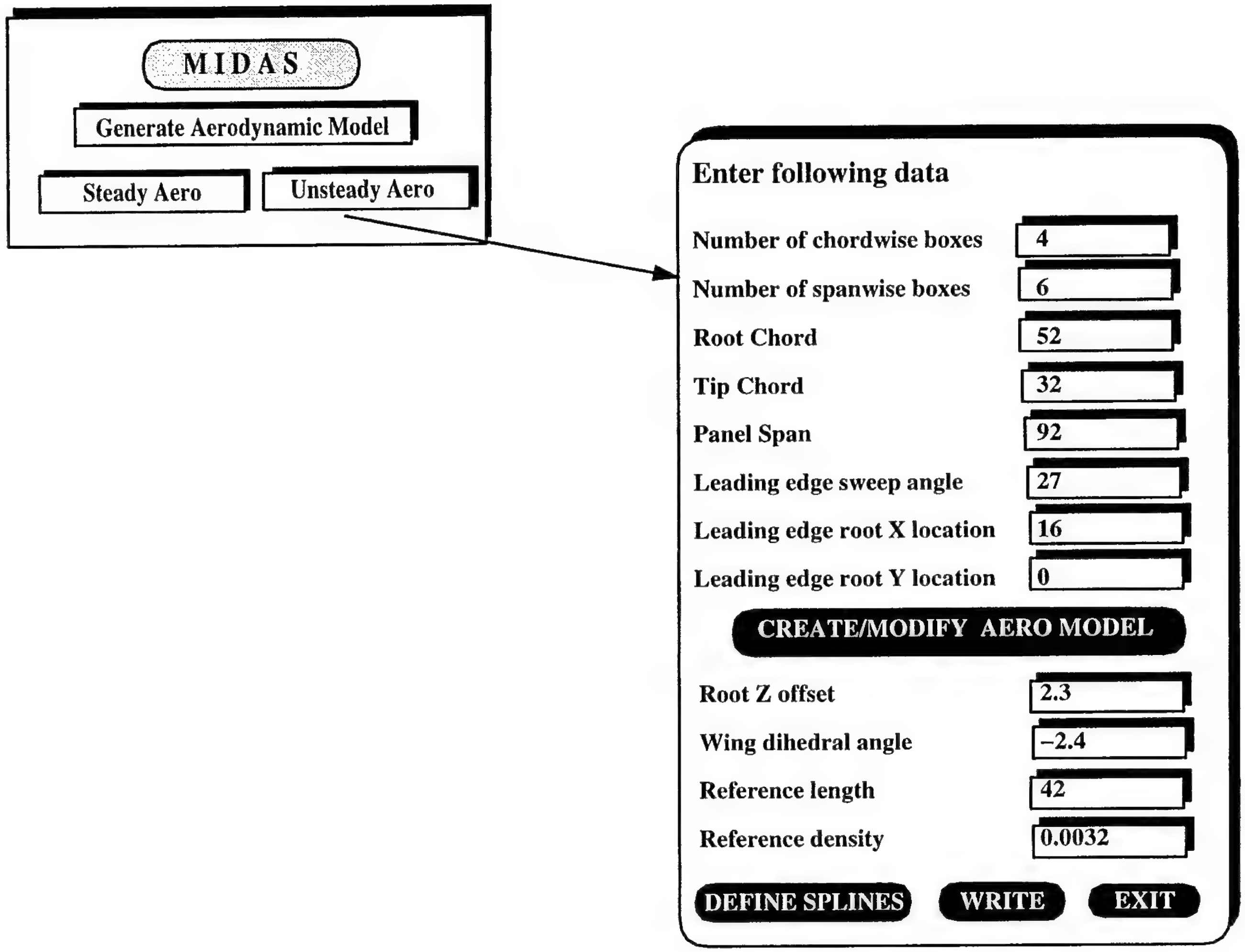


Figure 28. An Example of Open User Interface to Define Aerodynamic Parameters

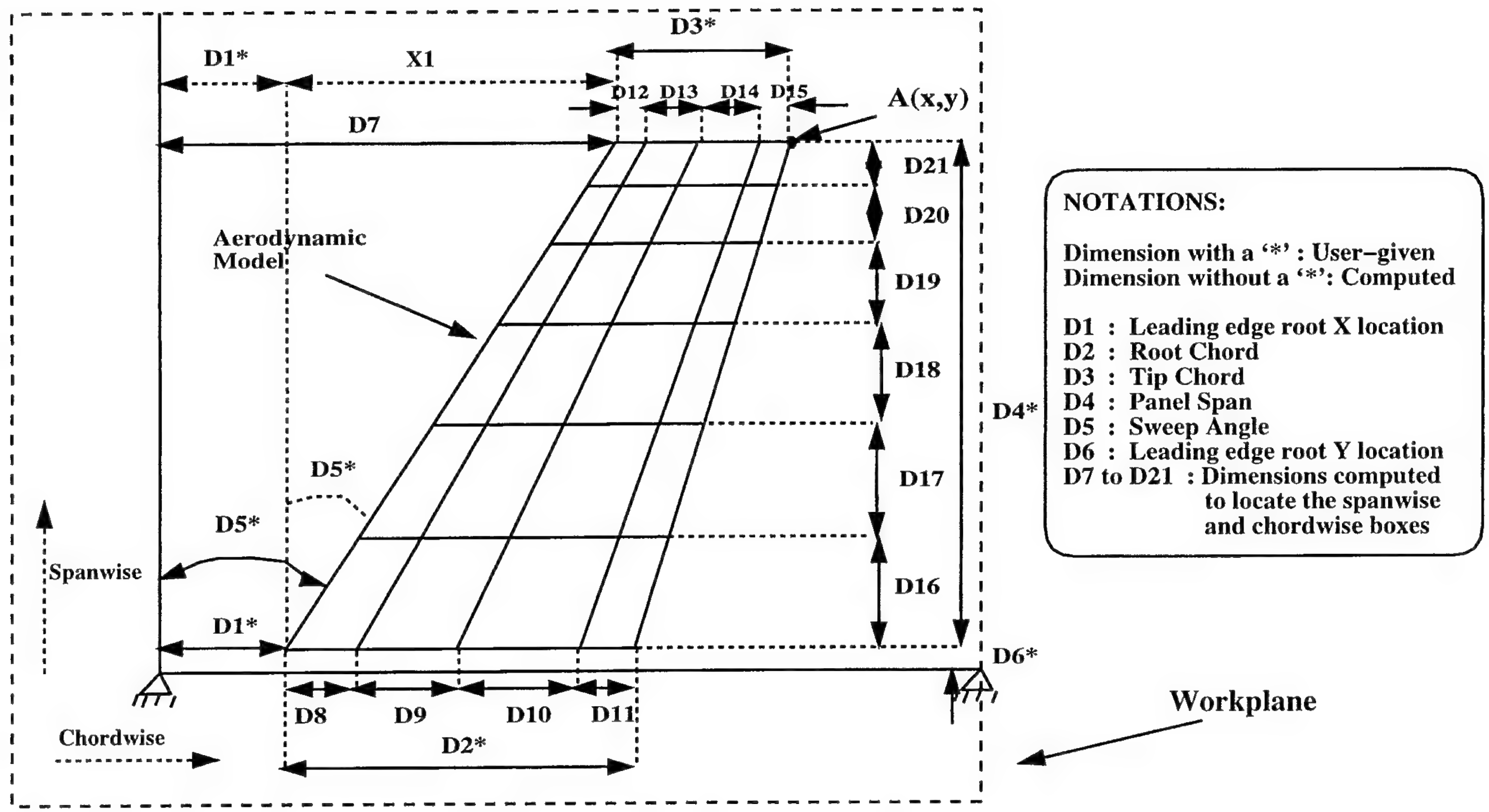
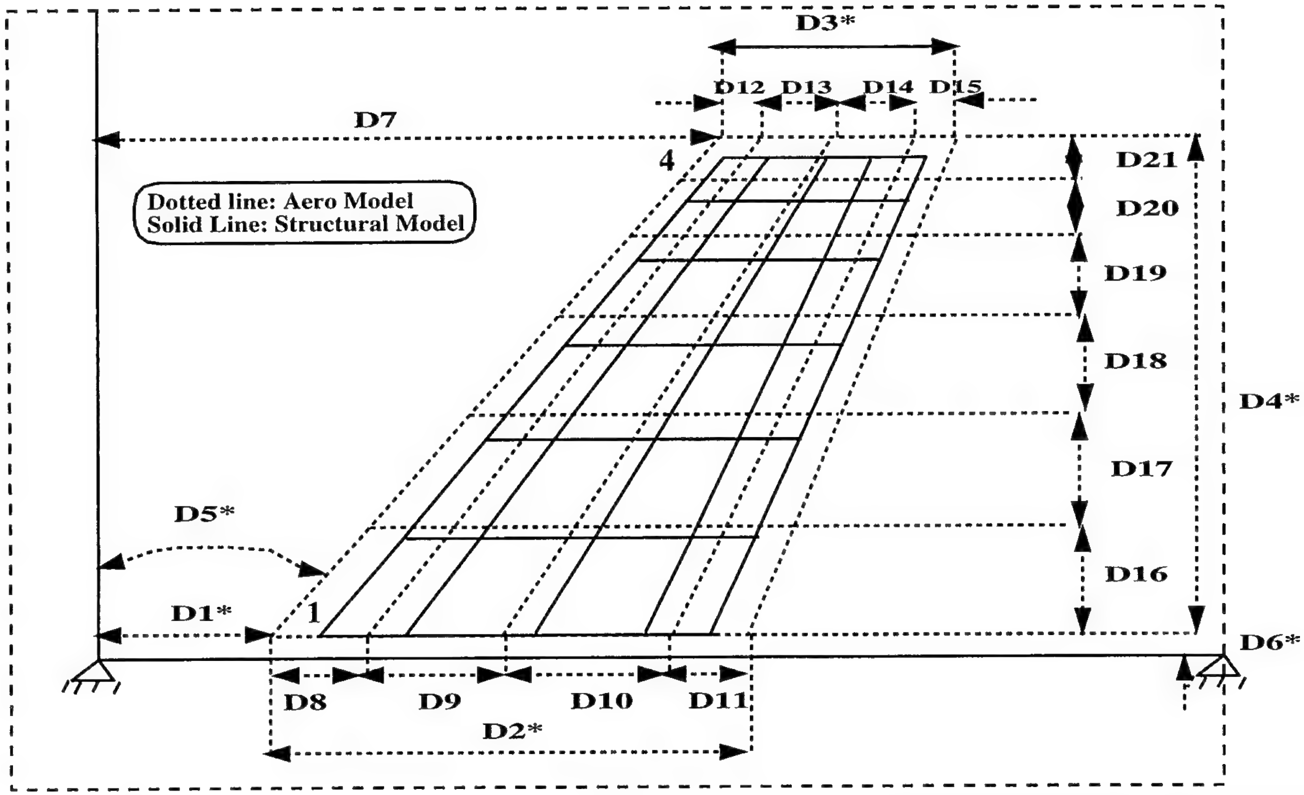


Figure 29. An Aero Model of a Wing Shown Along with the Dimensions Defined and Computed



CAERO1	1001				10	11	+1	
+1	16.	0.1	2.5	52.0	62.876	92.1	6.195	32.0
AEFACT	10	0.0	0.146	0.5	0.854	1.0		
AEFACT	11	0.0	0.259	0.5	0.707	0.866	0.966	1.0
AERO		42.0	0.0032					

Figure 30. Superimposition of an Aero Model on a Structural Model

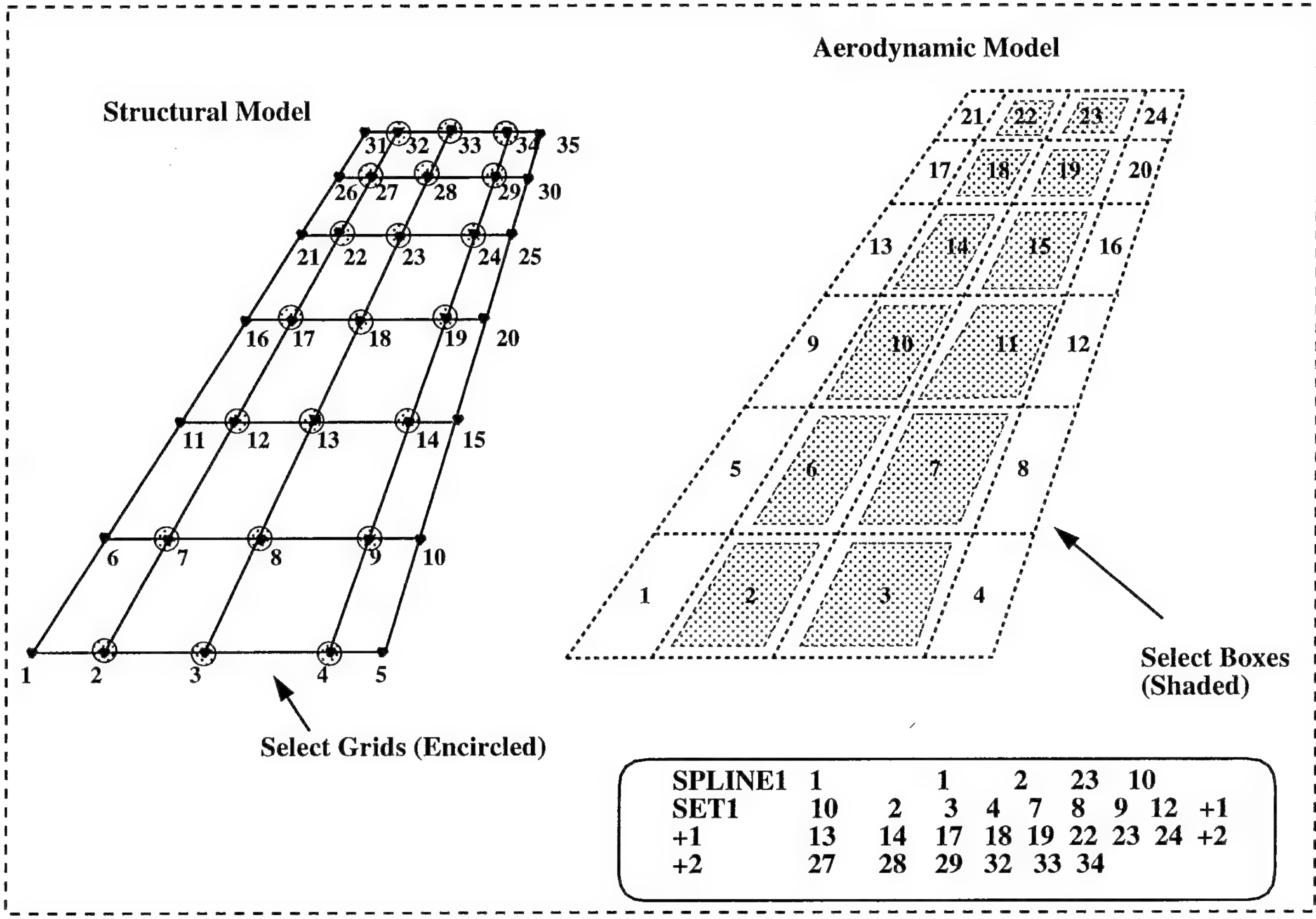


Figure 31. Aero and Structural Model Display For Spline Definition

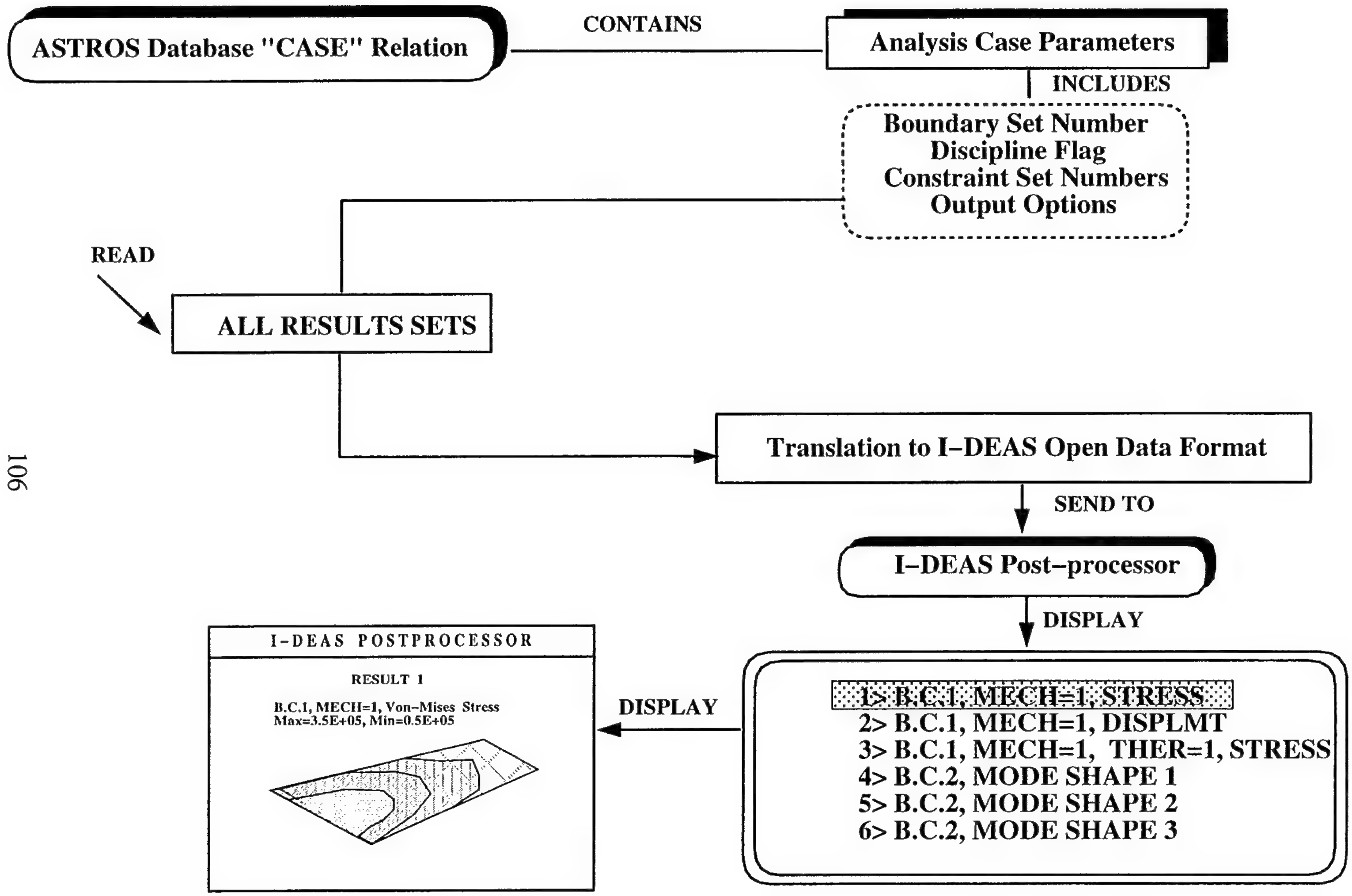


Figure 32. Steps Involved in Output Data Extraction in MIDAS

Chapter Four

Case Studies

This section presents example cases generated for ASTROS using MIDAS. Examples pertain to the analysis pre- and post-processing and design pre-processing modules of MIDAS. Cases include statics, modes and aero disciplines. Multiple boundary conditions, multiple load cases and multiple disciplines are considered in the example cases. All the data extracted, namely, element types, boundary condition types, load types, properties, design constraints, design variables, aerodynamic data, etc, are shown. Also different types of analysis sets generated for solution control are demonstrated. The examples include space truss, wing, fuselage and gas turbine blade structures.

Before illustrating the cases that were studied in MIDAS, general procedures for the installation and use of MIDAS are presented here:

MIDAS has been developed currently for the Silicon Graphics workstations and can be ported to other UNIX platforms. The presence of the I-DEAS software on the same platform is essential to run MIDAS since it uses the I-DEAS Open Architecture.

To start a session with MIDAS, the user has to first start an I-DEAS process from the directory in which the MIDAS executables reside. The MIDAS front end interface shown in Figure 33(a) comes up by default. The 'Analysis' and 'Optimization' options have menus as shown in Figure 33(b) and Figure 33(c). Options are provided to import an ASTROS file and run ASTROS from within MIDAS. (shown in Figure 33(d) and Figure 33(e) respectively). The 'Run Postprocessor' options queries the ASTROS database name and password as shown in Figure 33(f).

To generate a complete ASTROS input data deck, the 'Generate Complete Analysis File' and 'Generate Complete Design File' options can be selected as shown in Figure 33(b) and Figure 33(c). If the user wishes to modify only a certain section of the input, then the respective options shown in Figure 33(b) and Figure 33(c) can be selected. For example, if the user wishes to modify the load sets, then only the 'Load' option in Figure 33(b) can be selected or if the user wishes to include design variable information, then only the 'Design Variables' option shown in Figure 33(c) can be selected. This procedure has been applied to all the cases presented in this chapter, that were generated using MIDAS

4.1 Examples of I-DEAS to ASTROS Model Translation

Example 1 (Ten bar truss model to show MIDAS features):

This example presents a simple case which will essentially demonstrate the analysis pre-processing characteristics of MIDAS. The criteria for selecting a simple problem at first is to demonstrate certain key features of MIDAS instead of masking its capabilities with voluminous data. A static analysis is considered here for a ten bar truss problem. The finite element model is first created in I-DEAS as shown in Figure 34. The model has six nodes and ten rod elements made of aluminum with a Young's modulus of $10X10^6$ psi and a weight density of 0.1 lb/cu in. The initial cross-section area of the rod members is 15 sq. in. Only the initial analysis part is presented. This model can be further modified and designed for minimum weight.

The model has a restraint set in which all the degrees of freedom are restrained for nodes 5 and 6. A mechanical load set is applied which has a magnitude 10000 lb each on nodes 2 and 3. The ASTROS file generated for this I-DEAS model is included along with Figure 34.

Example 2 (Element types):

In this example, element types that were extracted for ASTROS from I-DEAS are illustrated. For this purpose, a model file was created as shown in Figure 35a containing all the element types. The elements labeled E1 to E15 refer to the following types:

E1: Linear beam

E2: Rod

E3: Thin shell linear quadrilateral

E4: Plane stress linear quadrilateral

E5: Membrane linear quadrilateral

E6: Thin shell linear triangle

E7: Membrane linear triangle

E8: Solid linear brick

E9: Solid parabolic brick

E10: Solid cubic brick

E11: Node to node translational spring

E12: Node to ground translational spring

E13: Node to node rotational spring

E14: Node to ground rotational spring

E15: Lumped mass

The ASTROS translation for the I-DEAS element types is included along with Figure 35b.

Example 3 (Physical property types):

Continuing with the same model used in Example 2 (shown in Figure 35a), the various property table types extracted for the elements are included with Figure 35c.

Example 4 (ACOSS model for modal analysis):

This example will demonstrate the modal analysis capabilities of MIDAS. A space truss example has been selected for this purpose. The finite element model shown in Figure 36 has 33 nodes and 113 rod elements made of graphite epoxy material with a Young's modulus of $18.5X10^6$ psi and a material density of 0.000142 $lb - sec^2/in^4$. The initial rod cross-section area is 10 sq. in.

A restraint set is created in which nodes 3, 4 and 6 have their translational degrees of freedom restrained. A boundary set for modal analysis is created in I-DEAS and the parameters for modal analysis are defined as shown in Figure 36. Given's method for modal analysis is selected (shaded in Figure 36) and dynamic reduction control data is specified ('YES' shaded) along with the highest frequency of interest being specified as 12 hz. MIDAS is then used to translate this model to ASTROS format which is included with Figure 36.

Example 5 (Fuselage model for multiple disciplines):

This example demonstrates the multidisciplinary analysis capability in MIDAS. A fuselage model is selected for this purpose as shown in Figure 37. An analysis set is created for ASTROS, which has both statics and modes disciplines specified. The fuselage is modeled using thin shell linear quadrilateral (CQUAD4) elements. The structure is metallic and is made of aluminum with a Young's modulus of $1.0X10^7$ psi, Poisson's ratio 0.3 and material density 0.1 lb/cu in. The ends of the fuselage are fully restrained. For the static discipline a mechanical load of 2000 lbs distributed uniformly over the fuselage is considered. A modal analysis is carried out using the Given's method. For this example, the 'Generate Complete Analysis File' option was selected in MIDAS as shown in Figure 37. Modal analysis parameters were generated in a manner similar to the one presented in the previous example. The translated ASTROS model is included along with Figure 37.

Example 6 (ICW model for multiple load cases and boundary conditions):

This example demonstrates multiple boundary conditions and multiple load cases in MIDAS. An Intermediate Complexity Wing (ICW) model is selected for this purpose as shown in Figure 38. The top and bottom skins of the wing are modeled using thin shell linear quadrilateral (CQUAD4) elements and thin shell linear triangular elements (CTRIA3). The substructure (spars and ribs) is modeled using shear elements (CSHEAR). Rod elements (CROD) are used to model the posts. The wing structure is metallic and is made of aluminum with a Young's modulus of $1.0X10^7$ psi, Poisson's ratio 0.3 and material density 0.1 lb/cu in. The following restraints, constraints, kinematic degrees of freedom and load sets were created:

Restraint set 1: Root clamped

Restraint set 2: Root ball jointed

Restraint set 3: Root clamped partly and ball jointed partly

Constraint set 1: X trans(55) - X trans(53) = 0

Constraint set 2: Y trans(31) - Y trans(33) = 0

Constraint set 3: Z trans(21) - Z trans(23) = 0

Kinematic degree of freedom set 1: X translation inactive for node 9

Kinematic degree of freedom set 2: X translation inactive for node 5

Kinematic degree of freedom set 3: X translation inactive for node 1

Load set 1: 1000 lb airload distributed uniformly on the top skin of the wing

Load set 2: 500 lb airload distributed uniformly on the top skin of the wing plus 1000 lb each on nodes 33 and 53, acting downwards

Load set 3: Gravity load

Load set 4: Combination of load sets 1 and 3

Temperature set 1: All nodes at 200 deg F

Using these sets, four boundary sets were created in I-DEAS:

Boundary set 1: Restraint set 3, Constraint set 1, Kinematic degree of freedom set 1, Load set 4, temperature set 1

Boundary set 2: Restraint set 1, Constraint set 2, Kinematic degree of freedom set 2, Load set 1, temperature set 1

Boundary set 3: Restraint set 1, Constraint set 2, Kinematic degree of freedom set 2, Load set 4, no temperature set

Boundary set 4: Restraint set 2, Constraint set 3, Kinematic degree of freedom set 3, Load set 2, no temperature set

These sets were then translated to ASTROS format which are included with Figure 38.

Example 7 (ICW model to show boundary grouping):

This example demonstrates how common boundary sets in I-DEAS were grouped. I-DEAS generates each boundary set separately. If this was translated literally into ASTROS, it would result in solving for repeated boundary sets. Instead, if these were to be grouped, it would result in saving computational time. This can be demonstrated by taking the previous example of the ICW shown in Figure 38 and its corresponding ASTROS file. When we scan the four boundary sets created in I-DEAS (in the previous example), it can be noticed that boundary sets 1 and 2 do not have identical boundary conditions. However, boundary set 2 and boundary set 3 have identical boundary conditions. They had to be created separately in I-DEAS, because it is not possible to apply the temperature set with one load case and not apply the set with another in the same boundary set in I-DEAS. Again, boundary set 4 does not have identical boundary conditions as set 1 or set 2 or set 3. If this were to be translated literally to ASTROS, with out grouping the common sets, it would result in a solution control which looks like:

```
BOUNDARY SPC = 3, MPC = 1, REDUCE = 1
        STATICS(MECH = 4, GRAV = 4 , THERMAL = 1)
        SUBTITLE = Boundary set 1
        LABEL = Load Case
        PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
        STRAIN =NONE, FORCE=NONE, LOAD=NONE
BOUNDARY SPC = 1, MPC = 2, REDUCE = 2
        STATICS(MECH = 1 , THERMAL = 1)
        SUBTITLE = Boundary set 2
        LABEL = Load Case
        PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
        STRAIN =NONE, FORCE=NONE, LOAD=NONE
BOUNDARY SPC = 1, MPC = 2, REDUCE = 2
        STATICS(MECH = 4, GRAV = 4)
        SUBTITLE = Boundary set 2
        LABEL = Load Case
        PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
        STRAIN =NONE, FORCE=NONE, LOAD=NONE
BOUNDARY SPC = 2, MPC = 3, REDUCE = 3
```

```
        STATICS(MECH = 2)
        SUBTITLE = Boundary set 3
        LABEL = Load Case
        PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL
```

It can be noticed in this case that ASTROS would end up setting up matrices twice for the same boundary condition (sets 2 and 3 shown above) but with different load sets which would definitely result in more computer time. But if this solution control is generated with MIDAS, the common boundary sets (2 and 3) would be grouped and the resulting solution control would look like:

```
BOUNDARY SPC = 3, MPC = 1, REDUCE = 1
        STATICS(MECH = 4, GRAV = 4 , THERMAL = 1)
        SUBTITLE = Boundary set 1
        LABEL = Load Case
        PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
        STRAIN =NONE, FORCE=NONE, LOAD=NONE
BOUNDARY SPC = 1, MPC = 2, REDUCE = 2
        STATICS(MECH = 1 , THERMAL = 1)
        SUBTITLE = Boundary set 2
        LABEL = Load Case
        PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
        STRAIN =NONE, FORCE=NONE, LOAD=NONE
        STATICS(MECH = 4, GRAV = 4)
        SUBTITLE = Boundary set 2
        LABEL = Load Case
        PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
        STRAIN =NONE, FORCE=NONE, LOAD=NONE
BOUNDARY SPC = 2, MPC = 3, REDUCE = 3
        STATICS(MECH = 2)
        SUBTITLE = Boundary set 3
        LABEL = Load Case
        PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
        STRAIN =NONE, FORCE=NONE, LOAD=NONE
```

Example 8 (Rectangular wing model to show model modifications and updates):

This example demonstrates how data can be modified in an ASTROS input stream without having to regenerate the model each time a modification is made. Only that segment of the input file which was modified will be regenerated. Consider the example of a simple rectangular wing shown in Figure 39. It is made of aluminum with a Young's modulus of $1X10^7$ psi and weight density of 0.1 lb/cu in and Poisson's ratio of 0.3. The wing skins are modeled with membrane elements (CQDMEM1), whereas the substructure is modeled with shear elements (CSHEAR) and the posts with rod elements (CROD). For Case 1, a static analysis is conducted in which a restraint set wherein the root is clamped is applied along with a load set of 500 lbs acting on the wing skin. Figure 39 shows the MIDAS user-interface wherein the user makes the selection of the input segments to be generated. For this case, the user can select the first button wherein the entire ASTROS file is generated as shown in Figure 39. Suppose a modification is made to the model, say, the root is ball jointed instead of clamped. Now the user needs to modify only the boundary conditions in the model (Case 2), and hence can click on the 'Boundary Condition' button as shown in Figure 39 and MIDAS will update the information in the ASTROS file. If the user wants to change the magnitude of the loads and add some additional loads (Case 3), for example, the magnitude of the air load from 500 lbs to 1000 lbs and point loads of 500 lb on two nodes, then the user can select 'Loads' as shown in Figure 39 and MIDAS will update only the load segment of the input. Now if the user wishes to add another discipline (Case 4), say Modes, then the user creates a new boundary set in I-DEAS for modal analysis and runs only the solution control part of ASTROS as shown in Figure 39. MIDAS will include this new discipline along with the others. The ASTROS input generated for the four cases is included along with Figure 39.

Example 9 (Gas turbine blade model):

This example demonstrates how MIDAS can be applied to components other than the aerospace structures like wing, fuselage, etc. Figure 40 shows a gas turbine blade. The blade material has a Young's modulus of $2.9X10^7$ psi, material density of $7.51X10^-4\ lb-sec^2/in^4$ and Poisson's ratio of 0.3. The root of the blade is clamped. A modal analysis boundary set is created for the blade in I-DEAS, and the ASTROS translation of the model and the output are included along with Figure 40.

Example 10 (Design variable definition):

The following few cases demonstrate the optimization capabilities in MIDAS. This example demonstrates the design variable extraction (cross-section area and thickness). Design variables are defined both by unique and physical linking.

Figure 41 shows a space truss (considered in example 4) which is made of 113 rod elements. The (unique) design variables are the cross-section areas of the rod elements. There are 113 design variables, each with a lower bound of 0.01 sq in, upper bound of 1000.0 sq in and an initial value of 1.0 sq in. These design variables were first generated in I-DEAS, and the corresponding ASTROS translation is included along with this example.

Figure 42 shows a simple rectangular wing model made of membrane elements (top and bottom skin) and thin shell linear quadrilateral elements (substructure). The thickness of the elements are considered as design variables. The elements (1, 2, 3, 4, 5, 6, 7 and 8) on the top and bottom skins of the wing have been linked and defined as one design variable (Initial value = 0.2 in, Lower bound = 0.05 in and Upper bound = 1.5 in). All the elements of the spars (9, 10, 11, 12, 13 and 14) have been linked and considered as the second design variable (Initial value = 0.1 in, Lower bound = 0.01 in and Upper bound = 1 in). The elements in the ribs (15, 16, 17, 18, 19 and 20) have been linked and defined as the third design variable (Initial value = 0.1 in, Lower bound = 0.01 in and Upper bound = 1 in). These design variables were linked using the I-DEAS optimization task and the ASTROS input translation is included along with this example.

Alternately, they can also be linked using the elements physical properties. ASTROS allows both direct element linking and linking through physical property simultaneously. For example, all the elements on the wing skin (1, 2, 3, 4, 5, 6, 7

and 8) can be linked using direct element linking (Design variable 1), the spar and rib elements can be linked using their physical property tables (Design variable 2 for physical property table 2 and Design variable 3 for physical property table 3). The ASTROS input translation is included along with this example.

Example 11 (Design constraint definition):

This example demonstrates the design constraint extraction in MIDAS. Constraint extraction for a single discipline or for multiple disciplines is shown in this example.

An example of defining a constraint for a single discipline is demonstrated through the space structure model considered in example 10 (Figure 41). The design model defined in the previous example already contains 113 design variables. A lower bound frequency constraint is specified on this structure as 2 hz for the first mode and 3 hz for the second mode. These constraints were defined in I-DEAS, and its corresponding ASTROS translation is included along with this example.

To demonstrate the definition of multiple constraints (eg. for stress and displacement), consider the example of a ten bar truss shown in Figure 43. This structure is subjected to both displacement and stress constraints simultaneously. These constraints are placed in a single set, say, set number 100. Nodes 1, 2, 3 and 4 are subject to a displacement constraint of 2 inches in the transverse direction (+Y and -Y directions) and each member of the truss is subject to a stress constraint of 25000 psi. The ASTROS translation for this example is also included.

A definition of constraints for multidisciplinary design is shown through the example of a wing model shown in Figure 44. The two disciplines considered for this wing include Linear Statics and Normal Modes. The wing is thus being designed considering the impact of multiple disciplines simultaneously. The wing has 55 CQUAD4 elements each of which has a stress constraint of 40000 psi placed on it. A natural frequency constraints of 6 Hz (for the first mode), 12 Hz (for the second mode) and 18 Hz (on the third mode) are specified. The problem thus has two constraint sets (set number 100 for stress constraints and set number 200 for frequency constraints).

The corresponding ASTROS translation is included alongwith this example.

Example 12 (Aero model generation):

This example demonstrates the aerodynamic model generated for a wing structure. The structure considered for this example is the Intermediate Complexity Wing (ICW) model (considered in example 6) shown in Figure 45. To develop an aero model for this wing, the first step is to define various parameters as shown in Figure 45. The aero model defined here has 4 boxes chordwise, 6 boxes spanwise, a sweep angle of 27 degrees, a tip and root chord of 32 in and 52 in respectively and a span length of 92 inches.

After defining the parameters, the 'CREATE/MODIFY' option is selected, and the aero model shown in Figure 46 is generated for the parameters defined. The chordwise and spanwise divisions have been computed by MIDAS based upon the parameters specified in Figure 45.

After the aerodynamic model is generated, it is overlayed on the structural model as shown in Figure 47. The next step is to define the surface spline to attach the aerodynamic model to the structural grid points. For this, the 'DEFINE SPLINE' option shown in Figure 45 is selected. This option lays the structural model and the aerodynamic model side by side as shown in Figure 48. Using this display, the aerodynamic boxes that need to be attached to the structural grids are selected through graphical picking. (The aero boxes are first selected and then the grid points are selected on the structural model). The ASTROS translation of this aerodynamic model along with the spline definition is also included.

Example 13 (Post-processing ASTROS results):

The following example demonstrates the post-processing capability in MIDAS. Outputs extracted in MIDAS for ASTROS include nodal displacements, element stresses, strain energy and mode shapes.

Consider the example of the Intermediate Complexity Wing (ICW) shown in Figure 38. The top and bottom skins of the wing are modeled using CQUAD4 elements. The wing substructure is modeled using CSHEAR and CROD elements. The wing structure is metallic and is made of aluminum. Both Linear Statics and Normal Modes analyses were conducted on the wing.

The root of the wing was clamped and for the linear statics discipline, an airload of 500 lb distributed uniformly on the top skin of the wing was considered. Displacements, stresses and strain energies were extracted for this structure. For normal modes analysis Given's method was selected, and the natural frequencies for the first three modes were extracted.

Figure 49 shows the nodal displacements extracted by MIDAS. The values of displacements along the wing skin are indicated in this figure. Figure 50 shows the Von Mises stresses for the load case considered, and Figure 51 shows the strain energy distribution. The numerical values of these quantities corresponding to each contour are indicated in these figures. Figures 52, 53 and 54 show the mode shapes for the first three modes of the structure.

4.2 Advantages of using MIDAS for ASTROS

MIDAS offers definite advantages to the ASTROS user in modeling aspects. The comparison here is between generating the model through graphical means and generating the model through traditional text oriented means. Practical structures are often complex and require long modeling time. Under such circumstances, an application like MIDAS can speed up and simplify the process to a great extent. This can be explained by means of an example that has been discussed in this section.

Consider the example of the fuselage model (Example 5). Through I-DEAS Master Modeler, the configuration of the fuselage can be developed and later meshed to generate the the finite element model. Alternately, the finite element model can be generated directly, by developing an initial pattern of elements around the circumference and copying the remaining elements. Either way, the model generation takes very few minutes in I-DEAS and involves activation of just one push button in MIDAS ('Generate Complete Analysis File'), which again takes very few minutes to translate the model. This method can be compared with the traditional text-oriented method. Typing in 100's of nodes, elements, forces, etc (as can be seen from the model included with Figure 37) would take hours and is again not error free.

Modifications in the models are made very easy in MIDAS. As can be seen from Figure 39 (Example 8), a simple logic was worked out in which the user could update only the modified segment of the model. When the user specifies the name of the ASTROS input file, say 'model', then MIDAS creates 'model.ex' for executive control segment, 'model.sc' for solution control segment, 'model.bc' for boundary condition segment, etc as shown in Figure 55. During the I-DEAS model translation to the ASTROS format, MIDAS stores different segments of the model in these files and the complete ASTROS file in 'model.inp' is shown in Figure 55. If any change was

made to the model, say, boundary conditions (Case 2 in Example 8) or load (Case 3 in Example 8), etc, then MIDAS updates only the files 'model.bc' (in Case 2) or 'model.lo' (Case 3) and thus avoids the regeneration of file 'model.inp'. The files generated thus can be accessed at any time in different sessions. For example, the designer may conclude a session with I-DEAS and may freeze a design contained in 'model.inp'. If for some reason, the configuration needs modification in the same 'model.inp' file, then in a separate session altogether, the designer can ask MIDAS to make the changes and updates to 'model.inp'. In this manner, the designer can generate as many configurations as necessary from a single I-DEAS graphic model and thus maintain a history of the configurations created.

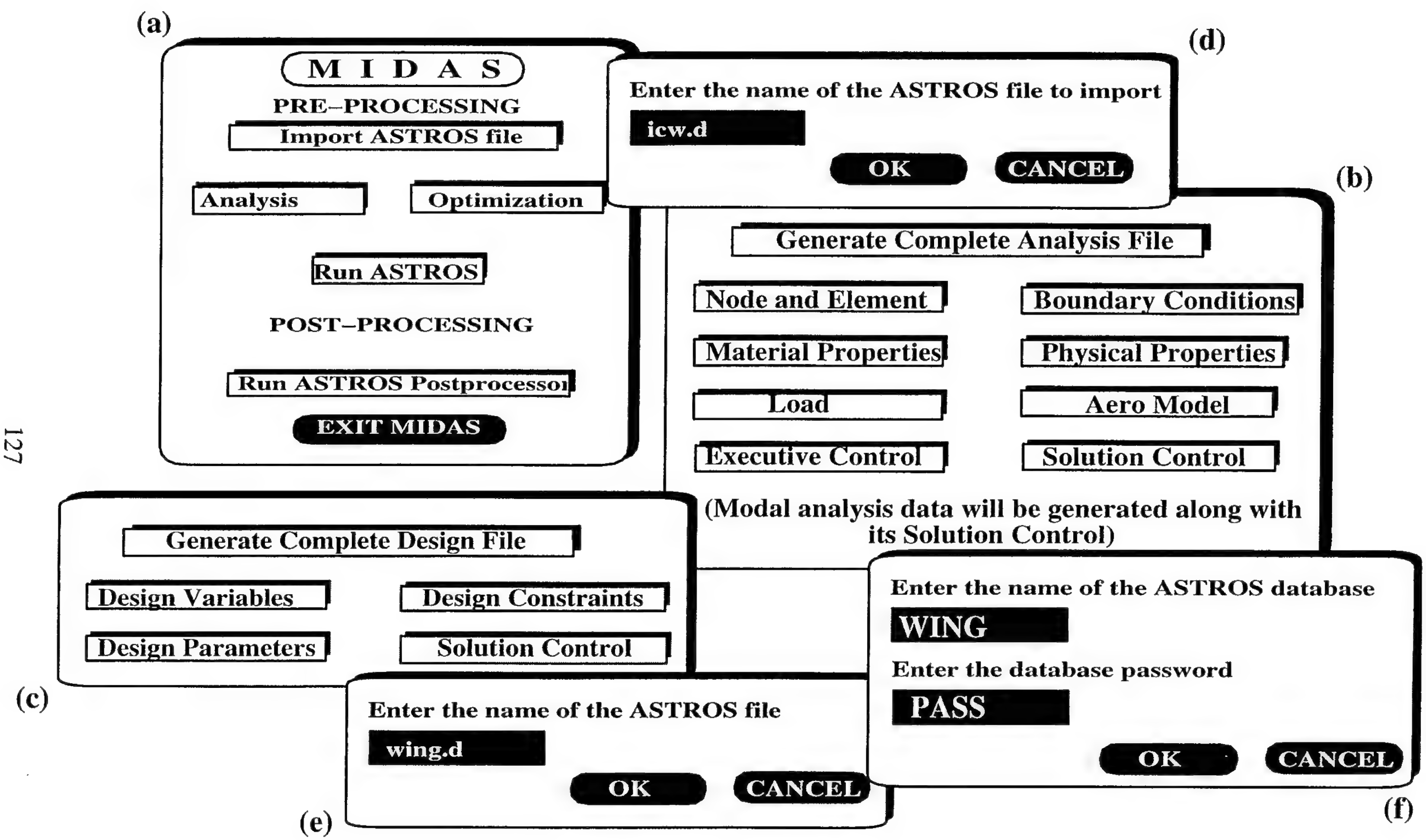


Figure 33. The MIDAS Layout

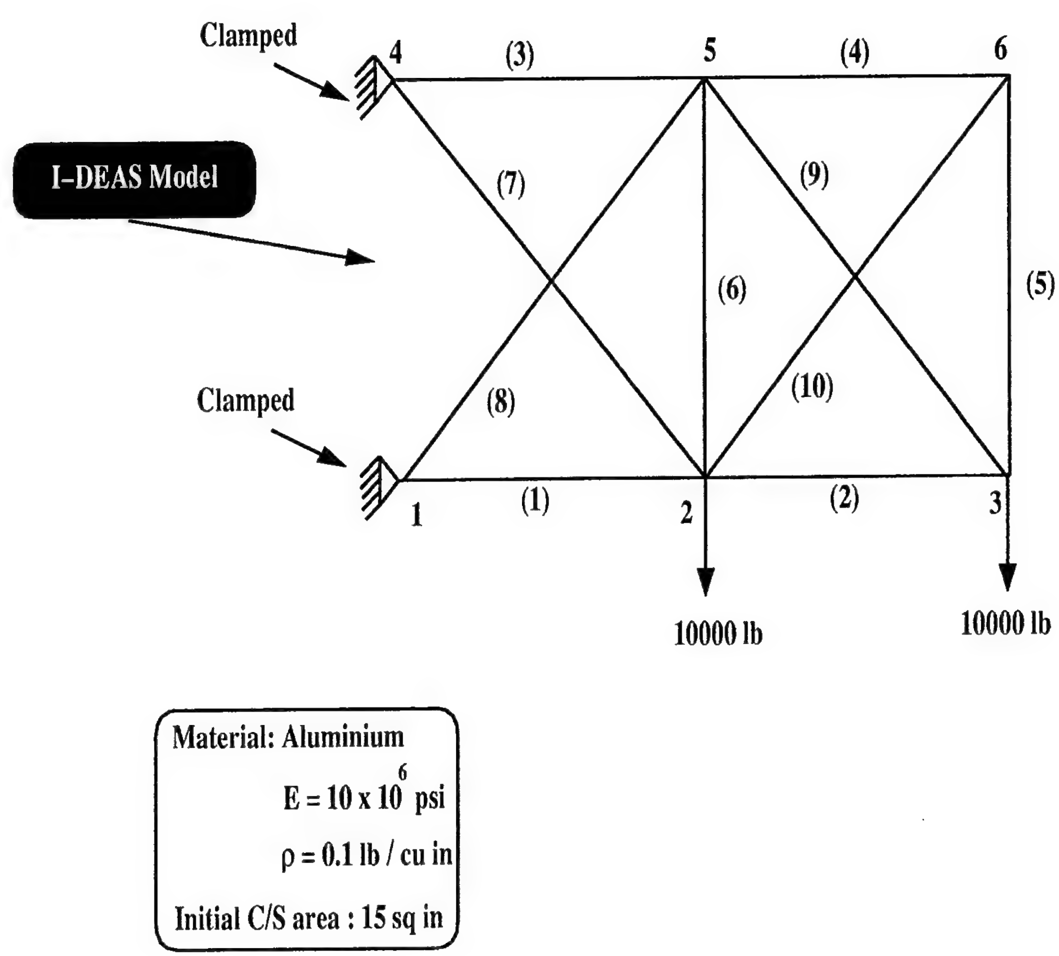


Figure 34. A Ten Bar Truss Model Created in I–DEAS for Example 1

ASTROS Input File for Tenbar Truss Structure (Example 1)

```
ASSIGN DATABASE TRUSS PASS NEW KEEP
SOLUTION
TITLE = Solution Control Packet
ANALYZE
BOUNDARY SPC = 1
   STATICS(MECH = 1)
   SUBTITLE = Boundary set 1
   LABEL = Load Case
   PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
       STRAIN =NONE, FORCE=NONE, LOAD=NONE
END
BEGIN BULK
$*****************************************
$                     NODE
$*****************************************
GRID   1          0.00   0.00   0.00
GRID   2          360.00 0.00   0.00
GRID   3          720.00 0.00   0.00
GRID   4          0.00   360.00 0.00
GRID   5          360.00 360.00 0.00
GRID   6          720.00 360.00 0.00
$*****************************************
$     ELEMENT (Rod)
$*****************************************
CROD   1    1    1    2
CROD   2    1    2    3
CROD   3    1    4    5
CROD   4    1    5    6
CROD   5    1    6    3
CROD   6    1    5    2
CROD   7    1    4    2
CROD   8    1    1    5
CROD   9    1    5    3
CROD   10   1    6    2
$*****************************************
$    BOUNDARY CONDITION (Restraints)
$*****************************************
SPC1   1    123456 1    4
$*****************************************
$       FORCE (Mechanical)
$*****************************************
FORCE  1    2         -1.0E+050.0   1.0   0.0
FORCE  1    3         -1.0E+050.0   1.0   0.0
$*****************************************
$       PHYSICAL PROPERTY
$*****************************************
PROD   1    2    15.00
```

```
$****************************************
$       MATERIAL PROPERTY
$****************************************
MAT1   2     1.0E+07       0.3    0.1
ENDDATA
```

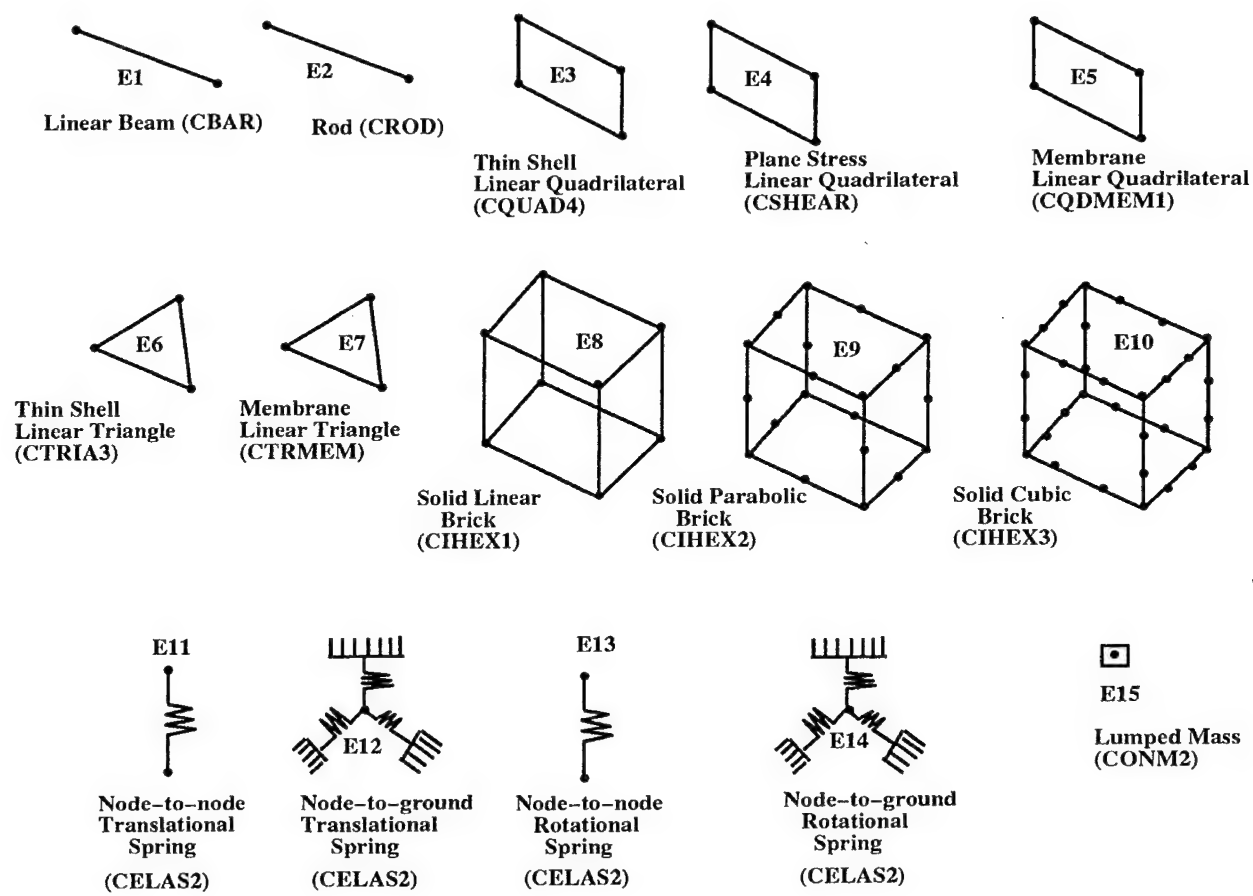


Figure 35a. List of I-DEAS Elements (in rectangular box) Translated to ASTROS Elements (in parenthesis)

Elem Number	I-DEAS Element	ASTROS Element
E1	Linear beam	CBAR
E2	Rod	CROD
E3	Thin shell linear quadrilateral	CQUAD4
E4	Plane stress linear quadrilateral	CSHEAR
E5	Membrane linear quadrilateral	CQDMEM1
E6	Thin shell linear triangle	CTRIA3
E7	Membrane linear triangle	CTRMEM
E8	Solid linear brick	CIHEX1
E9	Solid parabolic brick	CIHEX2
E10	Solid cubic brick	CIHEX3
E11	Node to node translational spring	CELAS2
E12	Node to ground translational spring	CELAS2
E13	Node to node rotational spring	CELAS2
E14	Node to ground rotational spring	CELAS2
E15	Lumped mass	CONM2

Figure 35b. List of I–DEAS Elements Translated to ASTROS Elements

ASTROS Elements Extracted from I-DEAS (Example 2)

```
$*******************************************
$                    LINEAR BEAM
$*******************************************
CBAR   1   1   1   2   0.0   1.0   0.0
$*******************************************
$                    ROD
$*******************************************
CROD   2   2   3   4
$*******************************************
$   THIN SHELL LINEAR QUADRILATERAL
$*******************************************
CQUAD4 3   3   7   8   6   5
$*******************************************
$   PLANE STRESS LINEAR QUADRILATERAL
$*******************************************
CSHEAR 4   4   11   12   10   9
$*******************************************
$   MEMBRANE LINEAR QUADRILATERAL
$*******************************************
CQDMEM1 5   5   15   16   14   13
$*******************************************
$   THIN SHELL LINEAR TRIANGLE
$*******************************************
CTRIA3 6   3   17   18   19
$*******************************************
$   MEMBRANE LINEAR TRIANGLE
$*******************************************
CTRMEM 7   5   20   21   22        1.0
$*******************************************
$   SOLID LINEAR BRICK
$*******************************************
CIHEX1 8   7   27   28   24   23   29   30   +8
+8   26   25
$*******************************************
$   SOLID PARABOLIC BRICK
$*******************************************
CIHEX2 9   7   35   40   36   47   32   44   +9
+9   31   49   39   42   46   43   37   41   +10
+10  38   48   34   45   33   50
$*******************************************
$   SOLID CUBIC BRICK
$*******************************************
CIHEX3 10   7   55   59   60   56   61   62   +10
+10  52   63   64   51   65   66   75   77   +11
+11  81   79   76   78   82   80   57   67   +12
+12  68   58   69   70   54   71   72   53   +13
+13  73   74
```

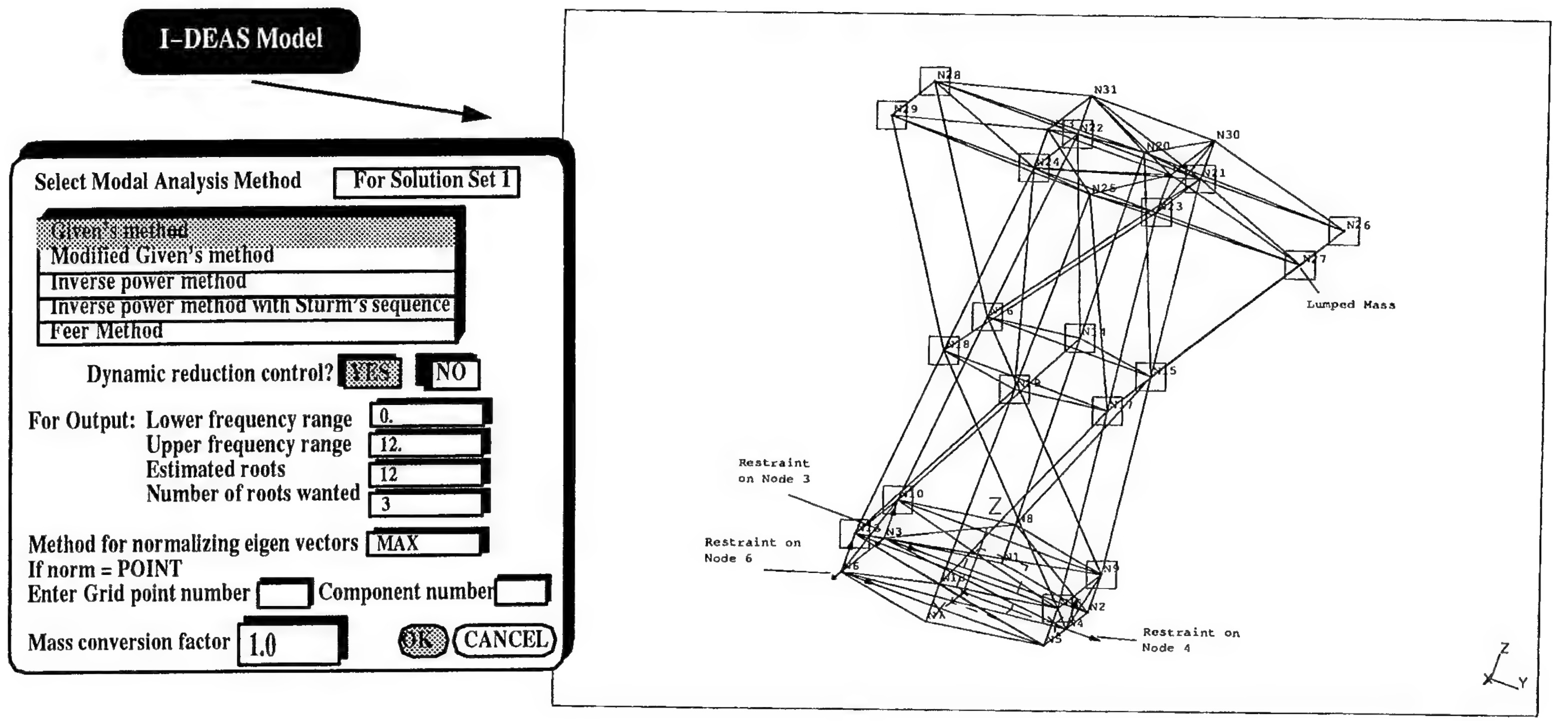
```
$*******************************************
$  NODE-NODE TRANSLATIONAL SPRING
$*******************************************
CELAS2 10001  6.28E+0389   1    87    1          0.100
CELAS2 10002  6.28E+0389   2    87    2          0.100
CELAS2 10003  6.28E+0389   3    87    3          0.100
$*******************************************
$  NODE-GROUND TRANSLATIONAL SPRING
$*******************************************
CELAS2 10004  6.28E+0388   1                   0.100
CELAS2 10005  6.28E+0388   2                   0.100
CELAS2 10006  6.28E+0388   3                   0.100
$*******************************************
$  NODE-NODE ROTATIONAL SPRING
$*******************************************
CELAS2 10007  6.28E+0386   4    84    4          0.100
CELAS2 10008  6.28E+0386   5    84    5          0.100
CELAS2 10009  6.28E+0386   6    84    6          0.100
$*******************************************
$  NODE-GROUND ROTATIONAL SPRING
$*******************************************
CELAS2 10010  6.28E+0385   4                   0.100
CELAS2 10011  6.28E+0385   5                   0.100
CELAS2 10012  6.28E+0385   6                   0.100
$*******************************************
$  LUMPED MASS
$*******************************************
CONM2  15     83         200.0000-10.00                +15
+15    1000.00        22500.00
```

ASTROS Property Type	For I–DEAS Element
PBAR	For Linear beam
PROD	For Rod
PSHELL	For Thin shell linear quadrilateral and Thin shell linear triangle
PSHEAR	For Plane stress linear quadrilateral
PQDMEM1	For Membrane linear quadrilateral
PTRMEM	For Membrane linear triangle
PIHEX	For linear, parabolic and cubic solid

Figure 35c. List of I–DEAS Property Types Translated to ASTROS Property Types

ASTROS Property Cards Extracted From I-DEAS (Example 3)

```
$*****************************************************
$                       LINEAR BEAM
$*****************************************************
PBAR   1    2    0.79   0.05   0.05   0.10   0.00        +15
+15   3.00   0.00
$*****************************************************
$                       ROD
$*****************************************************
PROD   2    1    0.79   0.10        0.00
$*****************************************************
$   THIN SHELL LINEAR QUADRILATERAL
$*****************************************************
PSHELL  3    2    0.10   2    1.00   2    0.83   100.00
$*****************************************************
$   PLANE STRESS LINEAR QUADRILATERAL
$*****************************************************
PSHEAR  4    2    0.20   100.00
$*****************************************************
$   MEMBRANE LINEAR QUADRILATERAL
$*****************************************************
PQDMEM1 5    2    0.10   100.00
$*****************************************************
$   THIN SHELL LINEAR TRIANGLE
$*****************************************************
PSHELL  3    2    0.10   2    1.00   2    0.83   100.00
$*****************************************************
$   MEMBRANE LINEAR TRIANGLE
$*****************************************************
PTRMEM  5    2    0.10   100.00
$*****************************************************
$  SOLID LINEAR, PARABOLIC AND CUBIC BRICK
$*****************************************************
PIHEX  7    1    2
```



Material: Graphite Epoxy

$E = 18.5 \times 10^{6}$ psi

$\rho = 0.000142$ lb / cu in

Initial C/S area : 10 sq in

Figure 36. Space Truss Model (ACOSS Structural Model) for Example 4

ASTROS Input File for Space Truss Structure (Example 4)

```
ASSIGN DATABASE ACOSS2 PASS NEW KEEP
SOLUTION
TITLE = Solution Control Packet
ANALYZE
BOUNDARY SPC = 1, METHOD = 1, DYNRED = 1
   MODES
   SUBTITLE = Boundary set 1
   LABEL = Modal Analysis
   PRINT ROOT = ALL, DISP(MODES=ALL)=ALL
END
BEGIN BULK
$*****************************************
$                         NODE
$*****************************************
GRID  1      -275.59 0.00   0.00
GRID  2      -157.48 196.85  0.00
GRID  3      -157.48 -196.85 0.00
GRID  4      0.00   196.85  0.00
GRID  5      157.48  196.85  0.00
GRID  6      157.48  -196.85 0.00
GRID  7      275.59  0.00   0.00
GRID  8      -275.59 0.00   78.74
GRID  9      -157.48 196.85  78.74
GRID  10     -157.48 -196.85 78.74
GRID  11     157.48  196.85  78.74
GRID  12     157.48  -196.85 78.74
GRID  13     275.59  0.00   78.74
GRID  14     -236.22 0.00   472.44
GRID  15     -157.48 157.48  472.44
GRID  16     -157.48 -157.48 472.44
GRID  17     157.48  157.48  472.44
GRID  18     157.48  -157.48 472.44
GRID  19     236.22  0.00   472.44
GRID  20     -196.85 0.00   866.14
GRID  21     -157.48 118.11  866.14
GRID  22     -157.48 -118.11 866.14
GRID  23     157.48  118.11  866.14
GRID  24     157.48  -118.11 866.14
GRID  25     196.85  0.00   866.14
GRID  26     -157.48 393.70  866.14
GRID  27     157.48  393.70  866.14
GRID  28     -157.48 -393.70 866.14
GRID  29     157.48  -393.70 866.14
GRID  30     -157.48 118.11  944.88
GRID  31     -157.48 -118.11 944.88
GRID  32     157.48  118.11  944.88
GRID  33     157.48  -118.11 944.88
```

```
$*****************************************
$                  ELEMENT (Rod)
$*****************************************
CROD  1   1   1   2
CROD  2   1   1   3
CROD  3   1   2   3
CROD  4   1   2   4
CROD  5   1   3   4
CROD  6   1   4   5
CROD  7   1   4   6
CROD  8   1   3   6
CROD  9   1   5   6
CROD  10  1   5   7
CROD  11  1   6   7
CROD  12  1   1   8
CROD  13  1   2   9
CROD  14  1   3   10
CROD  15  1   5   11
CROD  16  1   6   12
CROD  17  1   7   13
CROD  18  1   3   8
CROD  19  1   2   8
CROD  20  1   3   9
CROD  21  1   4   9
CROD  22  1   4   11
CROD  23  1   5   12
CROD  24  1   5   13
CROD  25  1   6   13
CROD  26  1   3   12
CROD  27  1   6   10
CROD  28  1   8   9
CROD  29  1   8   10
CROD  30  1   9   10
CROD  31  1   9   12
CROD  32  1   10  11
CROD  33  1   9   11
CROD  34  1   10  12
CROD  35  1   11  12
CROD  36  1   11  13
CROD  37  1   12  13
CROD  38  1   14  15
CROD  39  1   14  16
CROD  40  1   15  16
CROD  41  1   17  18
CROD  42  1   17  19
CROD  43  1   18  19
CROD  44  1   8   14
CROD  45  1   10  14
CROD  46  1   10  16
CROD  47  1   9   16
CROD  48  1   9   15
```

CROD	49	1	11	17
CROD	50	1	8	15
CROD	51	1	11	18
CROD	52	1	12	18
CROD	53	1	12	19
CROD	54	1	13	19
CROD	55	1	13	17
CROD	56	1	14	20
CROD	57	1	14	22
CROD	58	1	16	22
CROD	59	1	16	21
CROD	60	1	15	21
CROD	61	1	15	20
CROD	62	1	17	23
CROD	63	1	18	23
CROD	64	1	18	24
CROD	65	1	19	24
CROD	66	1	19	25
CROD	67	1	17	25
CROD	68	1	15	26
CROD	69	1	16	28
CROD	70	1	17	27
CROD	71	1	18	29
CROD	72	1	20	21
CROD	73	1	20	22
CROD	74	1	21	22
CROD	75	1	23	24
CROD	76	1	23	25
CROD	77	1	24	25
CROD	78	1	21	23
CROD	79	1	21	24
CROD	80	1	22	24
CROD	81	1	21	30
CROD	82	1	22	31
CROD	83	1	24	33
CROD	84	1	23	32
CROD	85	1	23	30
CROD	86	1	21	31
CROD	87	1	22	33
CROD	88	1	24	32
CROD	89	1	30	31
CROD	90	1	31	33
CROD	91	1	32	33
CROD	92	1	30	32
CROD	93	1	31	32
CROD	94	1	20	26
CROD	95	1	21	26
CROD	96	1	21	27
CROD	97	1	23	27
CROD	98	1	25	27
CROD	99	1	26	27

```
CROD    100   1    20    28
CROD    101   1    22    28
CROD    102   1    24    28
CROD    103   1    24    29
CROD    104   1    25    29
CROD    105   1    28    29
CROD    106   1    26    30
CROD    107   1    27    32
CROD    108   1    28    31
CROD    109   1    29    33
CROD    110   1    20    31
CROD    111   1    20    30
CROD    112   1    25    33
CROD    113   1    25    32
CONM2   114   9          2.85E+00
CONM2   115   10         2.85E+00
CONM2   116   11         2.85E+00
CONM2   117   12         2.85E+00
CONM2   118   14         4.60E-02
CONM2   119   15         9.70E-02
CONM2   120   16         9.70E-02
CONM2   121   17         9.70E-02
CONM2   122   18         9.70E-02
CONM2   123   19         4.60E-02
CONM2   124   21         2.14E+00
CONM2   125   22         2.14E+00
CONM2   126   23         2.14E+00
CONM2   127   24         2.14E+00
CONM2   128   26         2.85E+00
CONM2   129   27         2.85E+00
CONM2   130   28         1.43E+00
CONM2   131   29         1.43E+00
$*****************************************
$     BOUNDARY CONDITION (Restraints)
$*****************************************
SPC1   1    123   3    4    6
GRDSET                               456
$*****************************************
$       PHYSICAL PROPERTY
$*****************************************
PROD   1    2    10.0   16.49      0.00
$*****************************************
$       MATERIAL PROPERTY
$*****************************************
MAT1   2    1.85E+07 9.3E+06      1.4E-04
$*****************************************
$     MODAL ANALYSIS PARAMETERS
$*****************************************
CONVERT MASS   1.
EIGR   1    GIV   0.    12.   12    3
DYNRED 1     12.00
ENDDATA
```

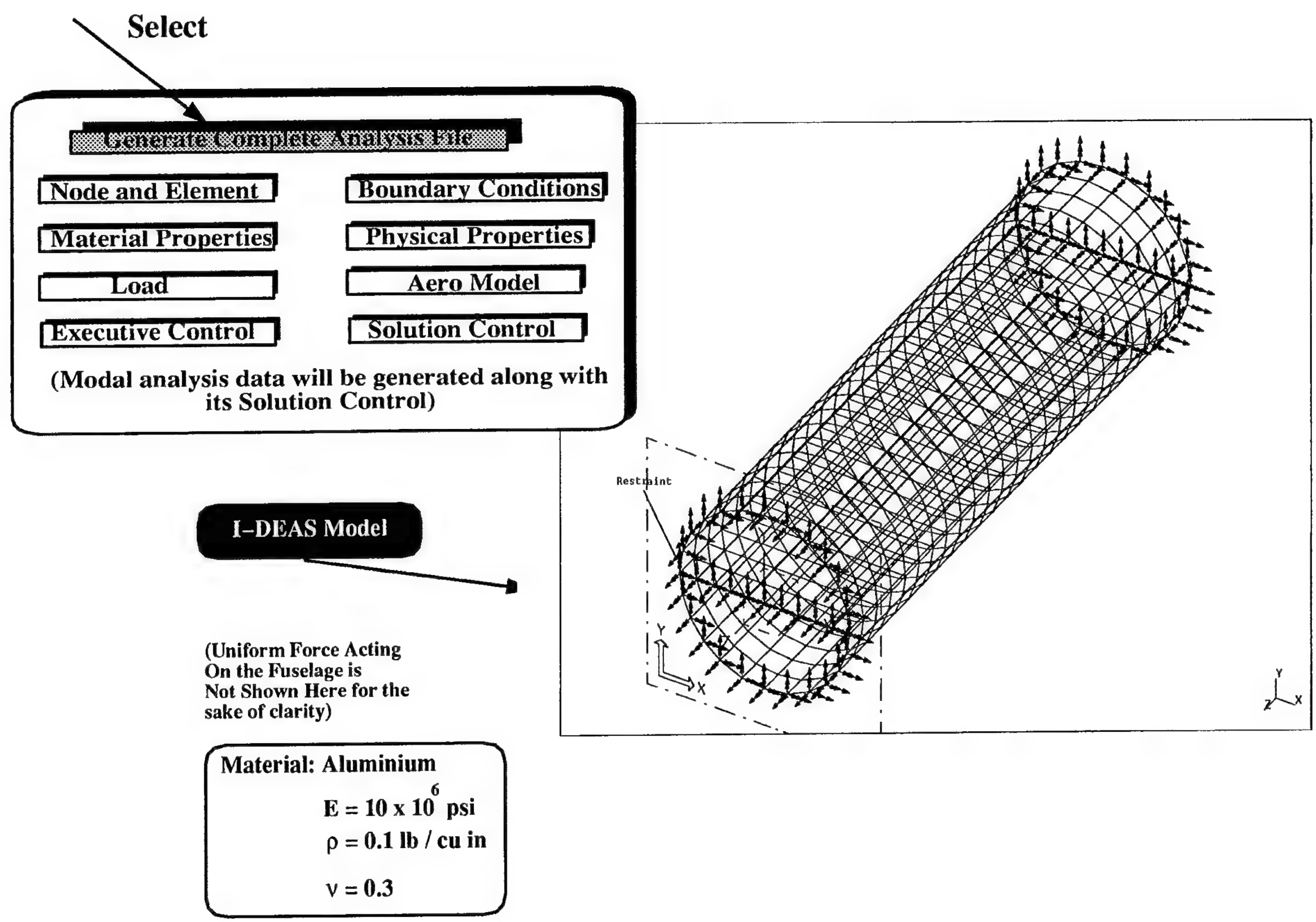


Figure 37. A Fuselage Model Created in I–DEAS for Example 5

ASTROS Input File for Fuselage (Example 5)

```
ASSIGN DATABASE FL PA NEW KEEP
SOLUTION
TITLE = Solution Control Packet
ANALYZE
BOUNDARY SPC = 1
   STATICS(MECH = 1)
   SUBTITLE = Boundary set 1
   LABEL = Load Case
   PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
       STRAIN =NONE, FORCE=NONE, LOAD=NONE
BOUNDARY SPC = 1, METHOD = 2
   MODES
   SUBTITLE = Boundary set 2
   LABEL = Modal Analysis
   PRINT ROOT = ALL, DISP(MODES=ALL)=ALL
END
BEGIN BULK
$*****************************************
$                    NODE
$*****************************************
GRID  1          0.00   72.00  0.00
GRID  2          -18.76 69.54  0.00
GRID  3          -37.59 61.44  0.00
GRID  4          -51.31 50.55  0.00
GRID  5          -62.68 35.48  0.00
GRID  6          -69.59 18.57  0.00
GRID  7          -72.00 0.00   0.00
GRID  8          -69.59 -16.50 0.00
GRID  9          -62.68 -33.42 0.00
GRID  10          -51.31 -48.48 0.00
......................................
......................................
GRID  824          44.00  0.00   -480.00
GRID  825          58.50  0.00   -480.00
$*****************************************
$ ELEMENT (Thin Shell Linear Quadrilateral)
$*****************************************
CQUAD4 1    1    1    2    35    34
CQUAD4 2    1    2    3    36    35
CQUAD4 3    1    3    4    37    36
..........................................
..........................................
CQUAD4 827  1    204   203   170   171
CQUAD4 828  1    72    71    38    39
```

```
$*******************************************
$    BOUNDARY CONDITION (Restraints)
$*******************************************
SPC1   1     123456 1    2    3    4    5    6    +7
+7    7    8    9    10   11   12   13   14   +15
+15   15   16   17   18   19   20   21   22   +23
+23   23   24   25   26   27   28   29   30   +31
+31   31   32   33   793  794  795  796  797  +39
+39   798  799  800  801  802  803  804  805  +47
+47   806  807  808  809  810  811  812  813  +55
+55   814  815  816  817  818  819  820  821  +63
+63   822  823  824  825
$*******************************************
$   FORCE  (Mechanical)
$*******************************************
FORCE  1    1        -2000.0 0.0   1.0   0.0
FORCE  1    2        -2000.0 0.0   1.0   0.0
FORCE  1    3        -2000.0 0.0   1.0   0.0
FORCE  1    4        -2000.0 0.0   1.0   0.0
.....................................
....................................
FORCE  1    825       -2000.0 0.0   1.0   0.0
$*******************************************
$        PHYSICAL PROPERTY
$*******************************************
PSHELL  1    2    1.00
$*******************************************
$        MATERIAL PROPERTY
$*******************************************
MAT1   2    1.0E+07      0.30   1.0E-01
$*******************************************
$      MODAL ANALYSIS PARAMETERS
$*******************************************
CONVERT MASS   1.
EIGR   2    GIV                3
ENDDATA
```

I–DEAS Model

(Temperature loading on the Wing is Not Shown Here for the sake of clarity)

Material: Aluminium

$E = 10 \times 10^6$ psi

$\rho = 0.1$ lb / cu in

$\nu = 0.3$

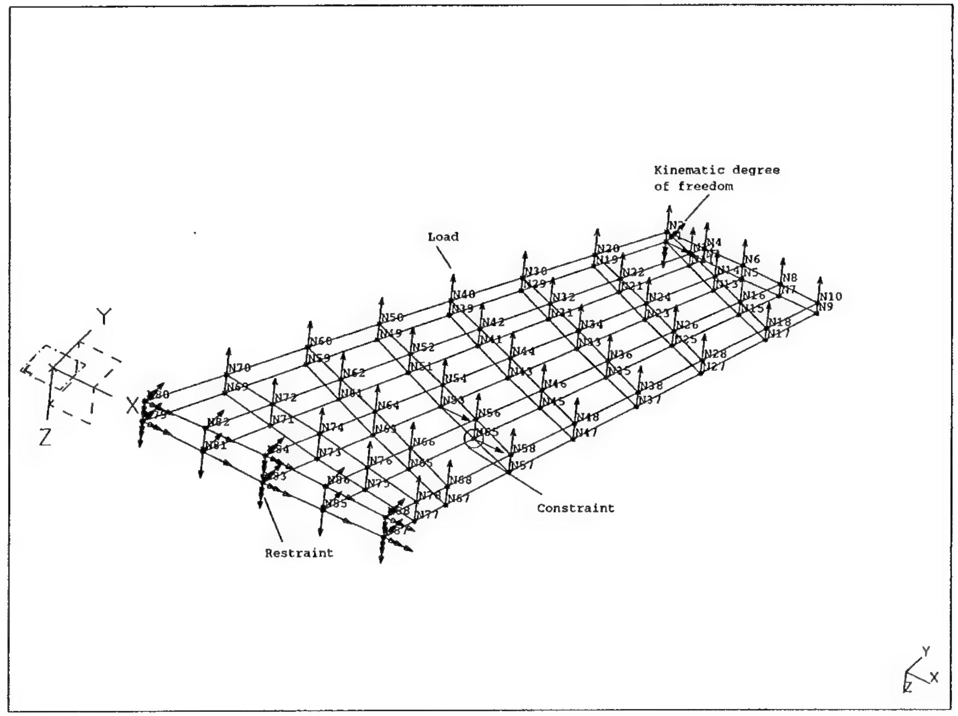


Figure 38. An IC Wing Model Created in I–DEAS

ASTROS Input File forICW Wing Structure (Example 6)

```
ASSIGN DATABASE ICW2 PASS NEW KEEP
SOLUTION
TITLE = Solution Control Packet
ANALYZE
BOUNDARY SPC = 3, MPC = 1, REDUCE = 1
   STATICS(MECH = 4, GRAV = 4 , THERMAL = 1)
   SUBTITLE = Boundary set 1
   LABEL = Load Case
   PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
       STRAIN =NONE, FORCE=NONE, LOAD=NONE
BOUNDARY SPC = 1, MPC = 2, REDUCE = 2
   STATICS(MECH = 1 , THERMAL = 1)
   SUBTITLE = Boundary set 2
   LABEL = Load Case
   PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
       STRAIN =NONE, FORCE=NONE, LOAD=NONE
   STATICS(MECH = 4, GRAV = 4)
   SUBTITLE = Boundary set 2
   LABEL = Load Case
   PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
       STRAIN =NONE, FORCE=NONE, LOAD=NONE
BOUNDARY SPC = 2, MPC = 3, REDUCE = 3
   STATICS(MECH = 2)
   SUBTITLE = Boundary set 3
   LABEL = Load Case
   PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
       STRAIN =NONE, FORCE=NONE, LOAD=NONE
END
BEGIN BULK
$*****************************************
$                          NODES
$*****************************************
GRID   1          63.50  90.00  1.12
GRID   2          63.50  90.00  -1.12
GRID   3          70.83  90.00  1.31
GRID   4          70.83  90.00  -1.31
GRID   5          78.17  90.00  1.50
GRID   6          78.17  90.00  -1.50
GRID   7          85.50  90.00  1.31
GRID   8          85.50  90.00  -1.31
GRID   9          92.83  90.00  1.12
GRID   10          92.83  90.00  -1.12
GRID   11          69.69  87.47  1.35
GRID   12          69.69  87.47  -1.35
GRID   13          76.10  84.85  1.59
GRID   14          76.10  84.85  -1.59
GRID   15          82.75  82.13  1.43
```

GRID	16	82.75	82.13	-1.43
GRID	17	89.65	79.31	1.26
GRID	18	89.65	79.31	-1.26
GRID	19	57.27	77.67	1.28
GRID	20	57.27	77.67	-1.28
GRID	21	63.99	74.92	1.53
GRID	22	63.99	74.92	-1.53
GRID	23	70.96	72.07	1.80
GRID	24	70.96	72.07	-1.80
GRID	25	78.19	69.12	1.62
GRID	26	78.19	69.12	-1.62
GRID	27	85.69	66.05	1.42
GRID	28	85.69	66.05	-1.42
GRID	29	51.03	65.34	1.43
GRID	30	51.03	65.34	-1.43
GRID	31	58.30	62.37	1.72
GRID	32	58.30	62.37	-1.72
GRID	33	65.83	59.29	2.01
GRID	34	65.83	59.29	-2.01
GRID	35	73.64	56.10	1.81
GRID	36	73.64	56.10	-1.81
GRID	37	81.74	52.79	1.59
GRID	38	81.74	52.79	-1.59
GRID	39	44.80	53.01	1.59
GRID	40	44.80	53.01	-1.59
GRID	41	52.60	49.82	1.90
GRID	42	52.60	49.82	-1.90
GRID	43	60.69	46.51	2.23
GRID	44	60.69	46.51	-2.23
GRID	45	69.08	43.08	2.00
GRID	46	69.08	43.08	-2.00
GRID	47	77.78	39.52	1.76
GRID	48	77.78	39.52	-1.76
GRID	49	38.56	40.68	1.74
GRID	50	38.56	40.68	-1.74
GRID	51	46.91	37.27	2.08
GRID	52	46.91	37.27	-2.08
GRID	53	55.56	33.73	2.44
GRID	54	55.56	33.73	-2.44
GRID	55	64.52	30.07	2.19
GRID	56	64.52	30.07	-2.19
GRID	57	73.83	26.26	1.92
GRID	58	73.83	26.26	-1.92
GRID	59	32.33	28.35	1.90
GRID	60	32.33	28.35	-1.90
GRID	61	41.21	24.72	2.27
GRID	62	41.21	24.72	-2.27
GRID	63	50.42	20.95	2.65
GRID	64	50.42	20.95	-2.65
GRID	65	59.97	17.05	2.38
GRID	66	59.97	17.05	-2.38

```
GRID   67      69.88  13.00  2.09
GRID   68      69.88  13.00  -2.09
GRID   69      25.17  14.17  2.07
GRID   70      25.17  14.17  -2.07
GRID   71      35.58  12.30  2.45
GRID   72      35.58  12.30  -2.45
GRID   73      46.18  10.40  2.83
GRID   74      46.18  10.40  -2.83
GRID   75      56.96  8.47   2.50
GRID   76      56.96  8.47   -2.50
GRID   77      67.94  6.50   2.17
GRID   78      67.94  6.50   -2.17
GRID   79      18.00  0.00   2.25
GRID   80      18.00  0.00   -2.25
GRID   81      30.00  0.00   2.62
GRID   82      30.00  0.00   -2.62
GRID   83      42.00  0.00   3.00
GRID   84      42.00  0.00   -3.00
GRID   85      54.00  0.00   2.62
GRID   86      54.00  0.00   -2.62
GRID   87      66.00  0.00   2.25
GRID   88      66.00  0.00   -2.25
$*******************************************
$                ELEMENT (Rod)
$*******************************************
CROD   1    4    1    2
CROD   2    4    3    4
CROD   3    4    5    6
CROD   4    4    7    8
CROD   5    4    9    10
CROD   6    4    11   12
CROD   7    4    13   14
CROD   8    4    15   16
CROD   9    4    17   18
CROD   10   4    19   20
CROD   11   4    21   22
CROD   12   4    23   24
CROD   13   4    25   26
CROD   14   4    27   28
CROD   15   4    29   30
CROD   16   4    31   32
CROD   17   4    33   34
CROD   18   4    35   36
CROD   19   4    37   38
CROD   20   4    39   40
CROD   21   4    41   42
CROD   22   4    43   44
CROD   23   4    45   46
CROD   24   4    47   48
CROD   25   4    49   50
CROD   26   4    51   52
```

```
CROD   27   4   53   54
CROD   28   4   55   56
CROD   29   4   57   58
CROD   30   4   59   60
CROD   31   4   61   62
CROD   32   4   63   64
CROD   33   4   65   66
CROD   34   4   67   68
CROD   35   4   69   70
CROD   36   4   71   72
CROD   37   4   73   74
CROD   38   4   75   76
CROD   39   4   77   78
$*****************************************
$  ELEMENT (Thin Shell Linear Triangle)
$*****************************************
CTRIA3 40   2   1   3   11
CTRIA3 41   2   2   4   12
$*****************************************
$ ELEMENT (Thin Shell Linear Quadrilateral)
$*****************************************
CQUAD4 42   1   3   5   13   11
CQUAD4 43   1   4   6   14   12
CQUAD4 44   1   5   7   15   13
CQUAD4 45   1   6   8   16   14
CQUAD4 46   1   7   9   17   15
CQUAD4 47   1   8   10  18   16
CQUAD4 48   1   1   11  21   19
CQUAD4 49   1   2   12  22   20
CQUAD4 50   1   11  13  23   21
CQUAD4 51   1   12  14  24   22
CQUAD4 52   1   13  15  25   23
CQUAD4 53   1   14  16  26   24
CQUAD4 54   1   15  17  27   25
CQUAD4 55   1   16  18  28   26
CQUAD4 56   1   19  21  31   29
CQUAD4 57   1   20  22  32   30
CQUAD4 58   1   21  23  33   31
CQUAD4 59   1   22  24  34   32
CQUAD4 60   1   23  25  35   33
CQUAD4 61   1   24  26  36   34
CQUAD4 62   1   25  27  37   35
CQUAD4 63   1   26  28  38   36
CQUAD4 64   1   29  31  41   39
CQUAD4 65   1   30  32  42   40
CQUAD4 66   1   31  33  43   41
CQUAD4 67   1   32  34  44   42
CQUAD4 68   1   33  35  45   43
CQUAD4 69   1   34  36  46   44
CQUAD4 70   1   35  37  47   45
CQUAD4 71   1   36  38  48   46
```

```
CQUAD4 72   1   39   41   51   49
CQUAD4 73   1   40   42   52   50
CQUAD4 74   1   41   43   53   51
CQUAD4 75   1   42   44   54   52
CQUAD4 76   1   43   45   55   53
CQUAD4 77   1   44   46   56   54
CQUAD4 78   1   45   47   57   55
CQUAD4 79   1   46   48   58   56
CQUAD4 80   1   49   51   61   59
CQUAD4 81   1   50   52   62   60
CQUAD4 82   1   51   53   63   61
CQUAD4 83   1   52   54   64   62
CQUAD4 84   1   53   55   65   63
CQUAD4 85   1   54   56   66   64
CQUAD4 86   1   55   57   67   65
CQUAD4 87   1   56   58   68   66
CQUAD4 88   1   59   61   71   69
CQUAD4 89   1   60   62   72   70
CQUAD4 90   1   61   63   73   71
CQUAD4 91   1   62   64   74   72
CQUAD4 92   1   63   65   75   73
CQUAD4 93   1   64   66   76   74
CQUAD4 94   1   65   67   77   75
CQUAD4 95   1   66   68   78   76
CQUAD4 96   1   69   71   81   79
CQUAD4 97   1   70   72   82   80
CQUAD4 98   1   71   73   83   81
CQUAD4 99   1   72   74   84   82
CQUAD4 100  1   73   75   85   83
CQUAD4 101  1   74   76   86   84
CQUAD4 102  1   75   77   87   85
CQUAD4 103  1   76   78   88   86
$****************************************
$ ELEMENT (Plane Stress Linear Quadrilateral)
$****************************************
CSHEAR 104  3   1    2    4    3
CSHEAR 105  3   3    4    6    5
CSHEAR 106  3   5    6    8    7
CSHEAR 107  3   7    8    10   9
CSHEAR 108  3   1    2    12   11
CSHEAR 109  3   11   12   14   13
CSHEAR 110  3   13   14   16   15
CSHEAR 111  3   15   16   18   17
CSHEAR 112  3   19   20   22   21
CSHEAR 113  3   21   22   24   23
CSHEAR 114  3   23   24   26   25
CSHEAR 115  3   25   26   28   27
CSHEAR 116  3   29   30   32   31
CSHEAR 117  3   31   32   34   33
CSHEAR 118  3   33   34   36   35
CSHEAR 119  3   35   36   38   37
```

```
CSHEAR 120   3    39    40    42    41
CSHEAR 121   3    41    42    44    43
CSHEAR 122   3    43    44    46    45
CSHEAR 123   3    45    46    48    47
CSHEAR 124   3    49    50    52    51
CSHEAR 125   3    51    52    54    53
CSHEAR 126   3    53    54    56    55
CSHEAR 127   3    55    56    58    57
CSHEAR 128   3    59    60    62    61
CSHEAR 129   3    61    62    64    63
CSHEAR 130   3    63    64    66    65
CSHEAR 131   3    65    66    68    67
CSHEAR 132   3    69    70    72    71
CSHEAR 133   3    71    72    74    73
CSHEAR 134   3    73    74    76    75
CSHEAR 135   3    75    76    78    77
CSHEAR 136   3    1     2     20    19
CSHEAR 137   3    19    20    30    29
CSHEAR 138   3    29    30    40    39
CSHEAR 139   3    39    40    50    49
CSHEAR 140   3    49    50    60    59
CSHEAR 141   3    59    60    70    69
CSHEAR 142   3    69    70    80    79
CSHEAR 143   3    5     6     14    13
CSHEAR 144   3    13    14    24    23
CSHEAR 145   3    23    24    34    33
CSHEAR 146   3    33    34    44    43
CSHEAR 147   3    43    44    54    53
CSHEAR 148   3    53    54    64    63
CSHEAR 149   3    63    64    74    73
CSHEAR 150   3    73    74    84    83
CSHEAR 151   3    9     10    18    17
CSHEAR 152   3    17    18    28    27
CSHEAR 153   3    27    28    38    37
CSHEAR 154   3    37    38    48    47
CSHEAR 155   3    47    48    58    57
CSHEAR 156   3    57    58    68    67
CSHEAR 157   3    67    68    78    77
CSHEAR 158   3    77    78    88    87
$*************************************************
$ BOUNDARY CONDITION (Restraints for Boundary Set 1)
$*************************************************
SPC1   3     123    81    82    85    86
SPC1   3     123456 79    80    83    84    87    88
$*************************************************
$ BOUNDARY CONDITION (Restraints for Boundary Set 2)
$*************************************************
SPC1   1     123456 79    80    81    82    83    84    +7
+7     85    86     87    88
```

```
$****************************************************
$ BOUNDARY CONDITION (Restraints for Boundary Set 3)
$****************************************************
SPC1  2    123  79   80   81   82   83   84   +7
+7    85   86   87   88
$************************************************************
$ BOUNDARY CONDITION (Kinematic Degree of Freedom for Boundary
$ Set 1)
$************************************************************
OMIT  1    9    1
$************************************************************
$ BOUNDARY CONDITION (Kinematic Degree of Freedom for Boundary
$ Set 2)
$************************************************************
OMIT  2    5    1
$************************************************************
$ BOUNDARY CONDITION (Kinematic Degree of Freedom for Boundary
$ Set 3)
$************************************************************
OMIT  3    1    1
$*****************************************************
$ BOUNDARY CONDITION (Constraints For Boundary Set 1)
$*****************************************************
MPC   1    55   1    1.00  53   1    -1.00
$*****************************************************
$ BOUNDARY CONDITION (Constraints For Boundary Set 2)
$*****************************************************
MPC   2    31   2    1.00  33   2    -1.00
$*****************************************************
$ BOUNDARY CONDITION (Constraints For Boundary Set 3)
$*****************************************************
MPC   3    21   3    1.00  23   3    -1.00
GRDSET                             456
$***************************************************
$ LOAD (Gravity, For Load Case 1 and Load Case 2)
$***************************************************
GRAV  4    0    1.0          3.5E+00
$***************************************************
$ LOAD (Mechanical, For Load Case 1 and Load Case 2)
$***************************************************
FORCE  4    2         -1000.0 0.0   0.0   1.0
FORCE  4    4         -1000.0 0.0   0.0   1.0
FORCE  4    6         -1000.0 0.0   0.0   1.0
FORCE  4    8         -1000.0 0.0   0.0   1.0
FORCE  4    10        -1000.0 0.0   0.0   1.0
FORCE  4    12        -1000.0 0.0   0.0   1.0
FORCE  4    14        -1000.0 0.0   0.0   1.0
FORCE  4    16        -1000.0 0.0   0.0   1.0
FORCE  4    18        -1000.0 0.0   0.0   1.0
FORCE  4    20        -1000.0 0.0   0.0   1.0
FORCE  4    22        -1000.0 0.0   0.0   1.0
```

```
FORCE  4   24      -1000.0 0.0   0.0   1.0
FORCE  4   26      -1000.0 0.0   0.0   1.0
FORCE  4   28      -1000.0 0.0   0.0   1.0
FORCE  4   30      -1000.0 0.0   0.0   1.0
FORCE  4   32      -1000.0 0.0   0.0   1.0
FORCE  4   34      -1000.0 0.0   0.0   1.0
FORCE  4   36      -1000.0 0.0   0.0   1.0
FORCE  4   38      -1000.0 0.0   0.0   1.0
FORCE  4   40      -1000.0 0.0   0.0   1.0
FORCE  4   42      -1000.0 0.0   0.0   1.0
FORCE  4   44      -1000.0 0.0   0.0   1.0
FORCE  4   46      -1000.0 0.0   0.0   1.0
FORCE  4   48      -1000.0 0.0   0.0   1.0
FORCE  4   50      -1000.0 0.0   0.0   1.0
FORCE  4   52      -1000.0 0.0   0.0   1.0
FORCE  4   54      -1000.0 0.0   0.0   1.0
FORCE  4   56      -1000.0 0.0   0.0   1.0
FORCE  4   58      -1000.0 0.0   0.0   1.0
FORCE  4   60      -1000.0 0.0   0.0   1.0
FORCE  4   62      -1000.0 0.0   0.0   1.0
FORCE  4   64      -1000.0 0.0   0.0   1.0
FORCE  4   66      -1000.0 0.0   0.0   1.0
FORCE  4   68      -1000.0 0.0   0.0   1.0
FORCE  4   70      -1000.0 0.0   0.0   1.0
FORCE  4   72      -1000.0 0.0   0.0   1.0
FORCE  4   74      -1000.0 0.0   0.0   1.0
FORCE  4   76      -1000.0 0.0   0.0   1.0
FORCE  4   78      -1000.0 0.0   0.0   1.0
$****************************************************
$ LOAD (Temperature, For Load Case 1 and Load Case 2)
$****************************************************
TEMP  1   1    200.00
TEMP  1   2    200.00
TEMP  1   3    200.00
TEMP  1   4    200.00
TEMP  1   5    200.00
TEMP  1   6    200.00
TEMP  1   7    200.00
TEMP  1   8    200.00
TEMP  1   9    200.00
TEMP  1   10   200.00
TEMP  1   11   200.00
TEMP  1   12   200.00
TEMP  1   13   200.00
TEMP  1   14   200.00
TEMP  1   15   200.00
TEMP  1   16   200.00
TEMP  1   17   200.00
TEMP  1   18   200.00
TEMP  1   19   200.00
TEMP  1   20   200.00
```

TEMP	1	21	200.00
TEMP	1	22	200.00
TEMP	1	23	200.00
TEMP	1	24	200.00
TEMP	1	25	200.00
TEMP	1	26	200.00
TEMP	1	27	200.00
TEMP	1	28	200.00
TEMP	1	29	200.00
TEMP	1	30	200.00
TEMP	1	31	200.00
TEMP	1	32	200.00
TEMP	1	33	200.00
TEMP	1	34	200.00
TEMP	1	35	200.00
TEMP	1	36	200.00
TEMP	1	37	200.00
TEMP	1	38	200.00
TEMP	1	39	200.00
TEMP	1	40	200.00
TEMP	1	41	200.00
TEMP	1	42	200.00
TEMP	1	43	200.00
TEMP	1	44	200.00
TEMP	1	45	200.00
TEMP	1	46	200.00
TEMP	1	47	200.00
TEMP	1	48	200.00
TEMP	1	49	200.00
TEMP	1	50	200.00
TEMP	1	51	200.00
TEMP	1	52	200.00
TEMP	1	53	200.00
TEMP	1	54	200.00
TEMP	1	55	200.00
TEMP	1	56	200.00
TEMP	1	57	200.00
TEMP	1	58	200.00
TEMP	1	59	200.00
TEMP	1	60	200.00
TEMP	1	61	200.00
TEMP	1	62	200.00
TEMP	1	63	200.00
TEMP	1	64	200.00
TEMP	1	65	200.00
TEMP	1	66	200.00
TEMP	1	67	200.00
TEMP	1	68	200.00
TEMP	1	69	200.00
TEMP	1	70	200.00
TEMP	1	71	200.00

```
TEMP    1     72     200.00
TEMP    1     73     200.00
TEMP    1     74     200.00
TEMP    1     75     200.00
TEMP    1     76     200.00
TEMP    1     77     200.00
TEMP    1     78     200.00
TEMP    1     79     200.00
TEMP    1     80     200.00
TEMP    1     81     200.00
TEMP    1     82     200.00
TEMP    1     83     200.00
TEMP    1     84     200.00
TEMP    1     85     200.00
TEMP    1     86     200.00
TEMP    1     87     200.00
TEMP    1     88     200.00
$***************************************
$     LOAD (Mechanical, Load Case 2)
$***************************************
FORCE  1     2          -1000.0 0.0    0.0    1.0
FORCE  1     4          -1000.0 0.0    0.0    1.0
FORCE  1     6          -1000.0 0.0    0.0    1.0
FORCE  1     8          -1000.0 0.0    0.0    1.0
FORCE  1     10         -1000.0 0.0    0.0    1.0
FORCE  1     12         -1000.0 0.0    0.0    1.0
FORCE  1     14         -1000.0 0.0    0.0    1.0
FORCE  1     16         -1000.0 0.0    0.0    1.0
FORCE  1     18         -1000.0 0.0    0.0    1.0
FORCE  1     20         -1000.0 0.0    0.0    1.0
FORCE  1     22         -1000.0 0.0    0.0    1.0
FORCE  1     24         -1000.0 0.0    0.0    1.0
FORCE  1     26         -1000.0 0.0    0.0    1.0
FORCE  1     28         -1000.0 0.0    0.0    1.0
FORCE  1     30         -1000.0 0.0    0.0    1.0
FORCE  1     32         -1000.0 0.0    0.0    1.0
FORCE  1     34         -1000.0 0.0    0.0    1.0
FORCE  1     36         -1000.0 0.0    0.0    1.0
FORCE  1     38         -1000.0 0.0    0.0    1.0
FORCE  1     40         -1000.0 0.0    0.0    1.0
FORCE  1     42         -1000.0 0.0    0.0    1.0
FORCE  1     44         -1000.0 0.0    0.0    1.0
FORCE  1     46         -1000.0 0.0    0.0    1.0
FORCE  1     48         -1000.0 0.0    0.0    1.0
FORCE  1     50         -1000.0 0.0    0.0    1.0
FORCE  1     52         -1000.0 0.0    0.0    1.0
FORCE  1     54         -1000.0 0.0    0.0    1.0
FORCE  1     56         -1000.0 0.0    0.0    1.0
FORCE  1     58         -1000.0 0.0    0.0    1.0
FORCE  1     60         -1000.0 0.0    0.0    1.0
FORCE  1     62         -1000.0 0.0    0.0    1.0
```

```
FORCE  1    64      -1000.0 0.0   0.0   1.0
FORCE  1    66      -1000.0 0.0   0.0   1.0
FORCE  1    68      -1000.0 0.0   0.0   1.0
FORCE  1    70      -1000.0 0.0   0.0   1.0
FORCE  1    72      -1000.0 0.0   0.0   1.0
FORCE  1    74      -1000.0 0.0   0.0   1.0
FORCE  1    76      -1000.0 0.0   0.0   1.0
FORCE  1    78      -1000.0 0.0   0.0   1.0
$**************************************************
$ LOAD (Mechanical, For Load Case 3)
$**************************************************
FORCE  2    2       -500.0 0.0    0.0   1.0
FORCE  2    4       -500.0 0.0    0.0   1.0
FORCE  2    6       -500.0 0.0    0.0   1.0
FORCE  2    8       -500.0 0.0    0.0   1.0
FORCE  2    10      -500.0 0.0    0.0   1.0
FORCE  2    12      -500.0 0.0    0.0   1.0
FORCE  2    14      -500.0 0.0    0.0   1.0
FORCE  2    16      -500.0 0.0    0.0   1.0
FORCE  2    18      -500.0 0.0    0.0   1.0
FORCE  2    20      -500.0 0.0    0.0   1.0
FORCE  2    22      -500.0 0.0    0.0   1.0
FORCE  2    24      -500.0 0.0    0.0   1.0
FORCE  2    26      -500.0 0.0    0.0   1.0
FORCE  2    28      -500.0 0.0    0.0   1.0
FORCE  2    30      -500.0 0.0    0.0   1.0
FORCE  2    32      -500.0 0.0    0.0   1.0
FORCE  2    33      1000.0 0.0    0.0   1.0
FORCE  2    34      -500.0 0.0    0.0   1.0
FORCE  2    36      -500.0 0.0    0.0   1.0
FORCE  2    38      -500.0 0.0    0.0   1.0
FORCE  2    40      -500.0 0.0    0.0   1.0
FORCE  2    42      -500.0 0.0    0.0   1.0
FORCE  2    44      -500.0 0.0    0.0   1.0
FORCE  2    46      -500.0 0.0    0.0   1.0
FORCE  2    48      -500.0 0.0    0.0   1.0
FORCE  2    50      -500.0 0.0    0.0   1.0
FORCE  2    52      -500.0 0.0    0.0   1.0
FORCE  2    53      1000.0 0.0    0.0   1.0
FORCE  2    54      -500.0 0.0    0.0   1.0
FORCE  2    56      -500.0 0.0    0.0   1.0
FORCE  2    58      -500.0 0.0    0.0   1.0
FORCE  2    60      -500.0 0.0    0.0   1.0
FORCE  2    62      -500.0 0.0    0.0   1.0
FORCE  2    64      -500.0 0.0    0.0   1.0
FORCE  2    66      -500.0 0.0    0.0   1.0
FORCE  2    68      -500.0 0.0    0.0   1.0
FORCE  2    70      -500.0 0.0    0.0   1.0
FORCE  2    72      -500.0 0.0    0.0   1.0
FORCE  2    74      -500.0 0.0    0.0   1.0
FORCE  2    76      -500.0 0.0    0.0   1.0
```

```
FORCE  2     78          -500.0 0.0    0.0    1.0
$*****************************************
$        PHYSICAL PROPERTY
$*****************************************
PROD   4     2     1.00   0.16          0.00
PSHELL 1     2     0.08
PSHEAR 3     2     1.00   0.00
PSHELL 2     2     0.08
$*****************************************
$        MATERIAL PROPERTY
$*****************************************
MAT1   2     1.0E+07       0.30   1.0E-01 6.5E-06 531.00
ENDDATA
```

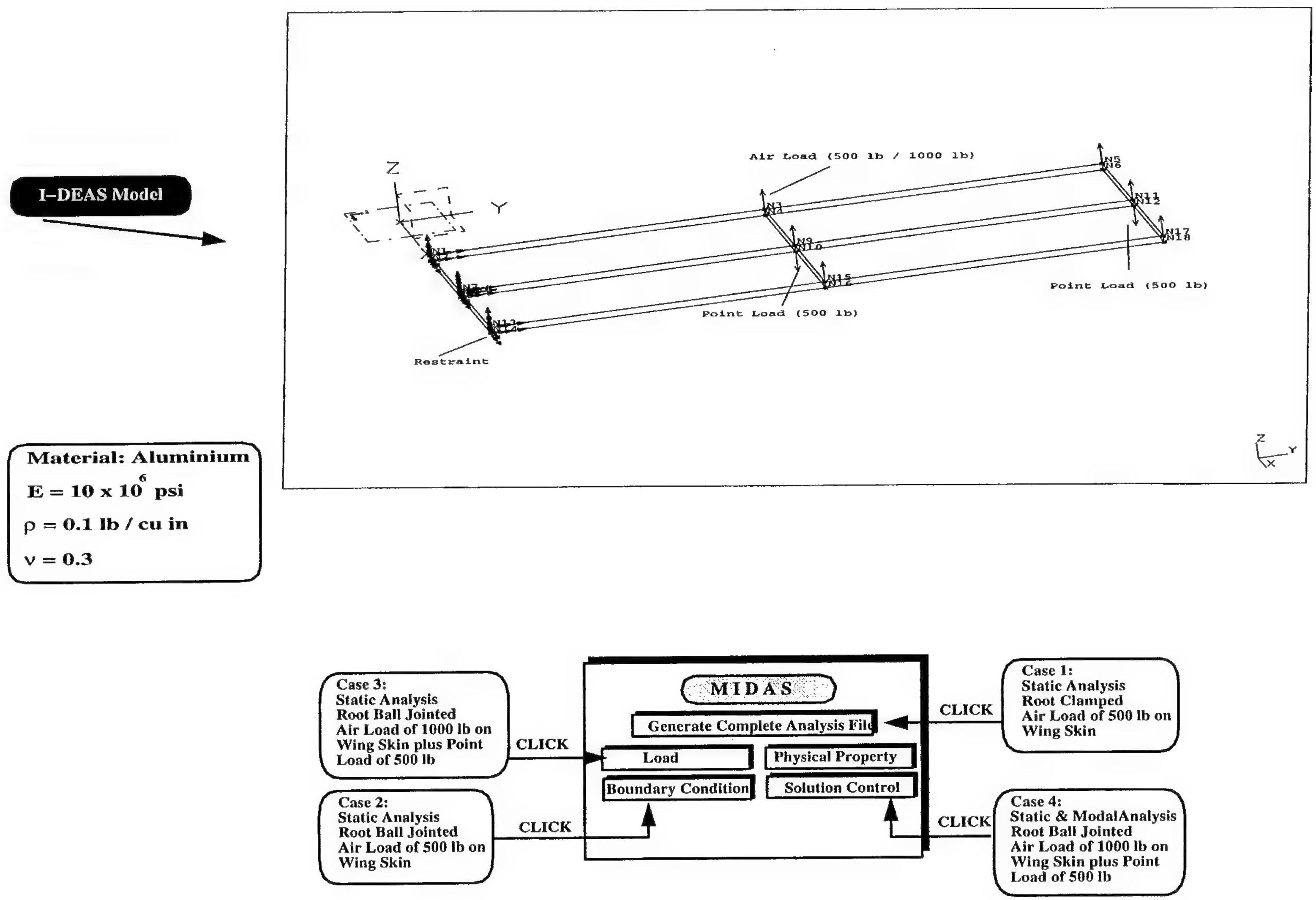


Figure 39. A Rectangular Wing Model Created in I–DEAS for Example 8

ASTROS Input File for Rectangular Wing Structure (Example 8)

(**Case 1:** Static Analysis, Root Clamped, Air load of 500 lb on wing skin)

```
ASSIGN DATABASE REC PASS NEW KEEP
SOLUTION
TITLE = Solution Control Packet
ANALYZE
BOUNDARY SPC = 1
   STATICS(MECH = 1)
   SUBTITLE = Boundary set 1
   LABEL = Load Case
   PRINT DISP=ALL, STRESS=ALL, ENERGY=ALL,
       STRAIN =NONE, FORCE=NONE, LOAD=NONE
END
BEGIN BULK
$*****************************************
$                         NODE
$*****************************************
GRID  1        10.00  0.00   0.50
GRID  2        10.00  0.00   -0.50
GRID  3        10.00  30.00  0.50
GRID  4        10.00  30.00  -0.50
GRID  5        10.00  60.00  0.50
GRID  6        10.00  60.00  -0.50
GRID  7        20.00  0.00   0.50
GRID  8        20.00  0.00   -0.50
GRID  9        20.00  30.00  0.50
GRID  10       20.00  30.00  -0.50
GRID  11       20.00  60.00  0.50
GRID  12       20.00  60.00  -0.50
GRID  13       30.00  0.00   0.50
GRID  14       30.00  0.00   -0.50
GRID  15       30.00  30.00  0.50
GRID  16       30.00  30.00  -0.50
GRID  17       30.00  60.00  0.50
GRID  18       30.00  60.00  -0.50
GRID  20       20.00  0.00   0.00
$*****************************************
$                ELEMENT (Rod)
$*****************************************
CROD  1   3   1   2
CROD  2   3   3   4
CROD  3   3   5   6
CROD  4   3   7   8
CROD  5   3   9   10
CROD  6   3   11  12
CROD  7   3   13  14
CROD  8   3   15  16
CROD  9   3   17  18
```

```
$*********************************************
$ ELEMENT (Plane Stress Linear Quadrilateral)
$*********************************************
CSHEAR  11   1   1   2   4   3
CSHEAR  12   1   3   4   6   5
$*********************************************
$ ELEMENT (Membrane Linear Quadrilateral)
$*********************************************
CQDMEM1 13   2   7   1   3   9
CQDMEM1 14   2   9   3   5   11
$*********************************************
$ ELEMENT (Plane Stress Linear Quadrilateral)
$*********************************************
CSHEAR  15   1   11  5   6   12
$*********************************************
$ ELEMENT (Membrane Linear Quadrilateral)
$*********************************************
CQDMEM1 16   2   10  4   6   12
CQDMEM1 17   2   8   2   4   10
$*********************************************
$ ELEMENT (Plane Stress Linear Quadrilateral)
CSHEAR  18   1   9   3   4   10
CSHEAR  19   1   15  9   10  16
$*********************************************
$ ELEMENT (Membrane Linear Quadrilateral)
$*********************************************
CQDMEM1 20   2   13  7   9   15
CQDMEM1 21   2   15  9   11  17
$*********************************************
$ ELEMENT (Plane Stress Linear Quadrilateral)
$*********************************************
CSHEAR  22   1   17  11  12  18
$*********************************************
$ ELEMENT (Membrane Linear Quadrilateral)
$*********************************************
CQDMEM1 23   2   16  10  12  18
$*********************************************
$ ELEMENT (Plane Stress Linear Quadrilateral)
CSHEAR  24   1   13  14  16  15
CSHEAR  25   1   15  16  18  17
$*********************************************
$ ELEMENT (Membrane Linear Quadrilateral)
$*********************************************
CQDMEM1 26   2   14  8   10  16
$*********************************************
$ ELEMENT (Thin Shell Linear Quadrilateral)
$*********************************************
CQUAD4  27   2   7   8   10  9
CQUAD4  28   2   9   10  12  11
CQUAD4  29   2   1   2   8   7
CQUAD4  30   2   7   8   14  13
```

```
$******************************************
$     BOUNDARY CONDITION (Restraints)
$******************************************
SPC1   1      123456 1     2     7     8     13     14     +7
+7     20
GRDSET                                   456
$     *     *     *     *     *     *     *     *     *
$******************************************
$        FORCE (Mechanical)
$******************************************
FORCE  1     3          500.0  0.0     0.0     1.0
FORCE  1     5          500.0  0.0     0.0     1.0
FORCE  1     9          500.0  0.0     0.0     1.0
FORCE  1     11         500.0  0.0     0.0     1.0
FORCE  1     15         500.0  0.0     0.0     1.0
FORCE  1     17         500.0  0.0     0.0     1.0
$******************************************
$        PHYSICAL PROPERTY
$******************************************
PROD   3     1     0.01   0.00        0.00
PSHELL 2     1     0.20
PSHEAR 1     1     0.05   0.00
PQDMEM1 2    1     0.20   0.00
$******************************************
$        MATERIAL PROPERTY
$******************************************
MAT1   1     1.0E+07 1.2E+07 0.3   0.1   6.5E-06 531.00
ENDDATA
```

ASTROS Input File for Rectangular Wing Structure (Example 8)

(Case 2: Static Analysis, Root Ball Jointed, Air load of 500 lb on wing skin)

```
ASSIGN DATABASE REC PASS NEW KEEP
SOLUTION
.........
.........
END
BEGIN BULK
$*****************************************
$                      NODE
$*****************************************
GRID   1         10.00  0.00   0.50
.......
GRID   20        20.00  0.00   0.00
$*****************************************
$               ELEMENT (Rod)
$*****************************************
CROD   1    3    1    2
.......
$**********************************************
$ ELEMENT (Thin Shell Linear Quadrilateral)
$**********************************************
...............................
CQUAD4 30   2    7    8    14   13
$*****************************************
$    BOUNDARY CONDITION (Restraints)
$*****************************************
SPC1   1    123  1    2    7    8    13   14   +7        Modified Section
+7     20
GRDSET                          456
$   *   *   *   *   *   *   *   *   *
$*****************************************
$      FORCE (Mechanical)
$*****************************************
..................
FORCE  1    17        500.0  0.0   0.0   1.0
$*****************************************
$      PHYSICAL PROPERTY
$*****************************************
PROD   3    1    0.01  0.00       0.00
......................
$*****************************************
$      MATERIAL PROPERTY
$*****************************************
MAT1   1    1.0E+07 1.2E+07 0.3   0.1   6.5E-06 531.00
ENDDATA
```

ASTROS Input File for Rectangular Wing Structure (Example 8)

(**Case 3:** Static Analysis, Root Ball Jointed, Air load of 1000 lb on wing skin plus point loads of 500 lb),

```
ASSIGN DATABASE REC PASS NEW KEEP
SOLUTION
.........
.........
END
BEGIN BULK
$******************************************
$                       NODE
$******************************************
GRID   1          10.00  0.00   0.50
.......
GRID   20         20.00  0.00   0.00
$******************************************
$               ELEMENT (Rod, Thin Shell Linear Quadrilateral)
$******************************************
CROD   1    3    1    2
.......
...............................
CQUAD4 30   2    7    8    14    13
$******************************************
$    BOUNDARY CONDITION (Restraints)
$******************************************
SPC1   1    123   1    2    7    8    13    14    +7
+7    20
$******************************************
$       FORCE (Mechanical)
$******************************************
FORCE 1 3     1000.0   0.0   0.0   1.0
FORCE 1 5     1000.0   0.0   0.0   1.0
FORCE 1 9     1000.0   0.0   0.0   1.0
FORCE 1 10    -500.0   0.0   0.0   1.0
FORCE 1 11    1000.0   0.0   0.0   1.0
FORCE 1 12    -500.0   0.0   0.0   1.0
FORCE 1 15    1000.0   0.0   0.0   1.0
FORCE 1 17    1000.0   0.0   0.0   1.0
$******************************************
$       PHYSICAL PROPERTY
$******************************************
PROD   3    1    0.01   0.00       0.00
.....................
$******************************************
$       MATERIAL PROPERTY
$******************************************
MAT1   1    1.0E+07 1.2E+07 0.3   0.1   6.5E-06 531.00
ENDDATA
```

Modified Section

ASTROS Input File for Rectangular Wing Structure (Example 8)

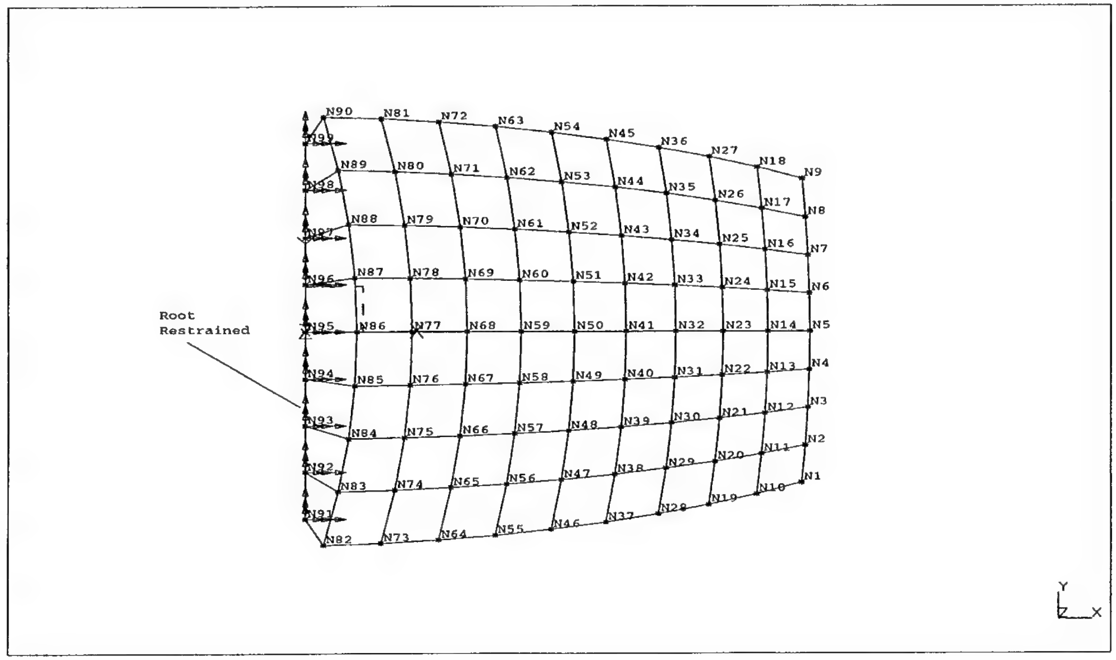
(**Case 4:** Static & ModalAnalysis, Root Ball Jointed, Air load of 1000 lb on wing skin plus point loads of 500 lb),

```
ASSIGN DATABASE REC PASS NEW KEEP
SOLUTION
.........
BOUNDARY SPC = 1, METHOD = 2
  MODES
  SUBTITLE = Boundary set 2
  LABEL = Modal Analysis
  PRINT ROOT = ALL, DISP(MODES=ALL)=ALL
END
BEGIN BULK
$*****************************************
$                        NODE
$*****************************************
........................
GRID   20          20.00  0.00   0.00
$*****************************************
$               ELEMENT (Rod, Thin Shell Linear Quadrilateral)
$*****************************************
CROD  1    3    1    2
...............................
CQUAD4 30   2    7    8    14    13
$*****************************************
$    BOUNDARY CONDITION (Restraints)
$*****************************************
SPC1  1    123   1    2    7    8    13    14    +7
+7    20
$*****************************************
$       FORCE (Mechanical)
$*****************************************
....................................
FORCE 1 17     1000.0  0.0    0.0    1.0
$*****************************************
$       PHYSICAL PROPERTY
$*****************************************
PROD  3    1    0.01  0.00        0.00
$*****************************************
$       MATERIAL PROPERTY
$*****************************************
MAT1  1    1.0E+07 1.2E+07 0.3   0.1  6.5E-06 531.00
$*****************************************
$      MODAL ANALYSIS PARAMETERS
$*****************************************
CONVERT MASS 1.0
EIGR 2 GIV
```

Modified Section

Modified Section

I–DEAS Model



$$E = 2.9 \times 10^{7} \text{ psi}$$
$$r = 7.51 \times 10^{-4} \text{ lb} - \text{sec}^{2} / \text{in}^{4}$$

Figure 40. A Gas Turbine Blade Model Created in I–DEAS for Example 9

ASTROS Input File for Gas Turbine Blade (Example 9)

```
ASSIGN DATABASE BL PASS NEW KEEP
SOLUTION
TITLE = Solution Control Packet
ANALYZE
BOUNDARY SPC = 1, METHOD = 1
   MODES
   SUBTITLE = Boundary set 1
   LABEL = Modal Analysis
   PRINT ROOT = ALL, DISP(MODES=ALL)=ALL
END
BEGIN BULK
$****************************************
$                     NODE
$****************************************
GRID  1      1.80  -0.71  1.35
GRID  2      1.81  -0.53  1.38
GRID  3      1.82  -0.35  1.40
GRID  4      1.83  -0.18  1.41
GRID  5      1.83  0.00   1.41
GRID  6      1.83  0.18   1.41
GRID  7      1.82  0.35   1.40
GRID  8      1.81  0.53   1.38
GRID  9      1.80  0.71   1.35
GRID  10     1.64  -0.76  1.11
GRID  11     1.66  -0.57  1.14
GRID  12     1.67  -0.38  1.15
GRID  13     1.68  -0.19  1.16
GRID  14     1.68  0.00   1.17
GRID  15     1.68  0.19   1.16
GRID  16     1.67  0.38   1.15
GRID  17     1.66  0.57   1.14
GRID  18     1.64  0.76   1.11
GRID  19     1.47  -0.81  0.89
GRID  20     1.49  -0.61  0.91
GRID  21     1.51  -0.40  0.93
GRID  22     1.51  -0.20  0.94
GRID  23     1.52  0.00   0.94
GRID  24     1.51  0.20   0.94
GRID  25     1.51  0.40   0.93
GRID  26     1.49  0.61   0.91
GRID  27     1.47  0.81   0.89
GRID  28     1.28  -0.85  0.68
GRID  29     1.31  -0.64  0.70
GRID  30     1.33  -0.43  0.72
GRID  31     1.34  -0.21  0.73
GRID  32     1.35  0.00   0.73
GRID  33     1.34  0.21   0.73
GRID  34     1.33  0.43   0.72
```

GRID	35	1.31	0.64	0.70
GRID	36	1.28	0.85	0.68
GRID	37	1.09	-0.89	0.50
GRID	38	1.13	-0.67	0.52
GRID	39	1.15	-0.45	0.53
GRID	40	1.16	-0.22	0.54
GRID	41	1.17	0.00	0.54
GRID	42	1.16	0.22	0.54
GRID	43	1.15	0.45	0.53
GRID	44	1.13	0.67	0.52
GRID	45	1.09	0.89	0.50
GRID	46	0.89	-0.92	0.34
GRID	47	0.93	-0.69	0.36
GRID	48	0.96	-0.46	0.37
GRID	49	0.97	-0.23	0.38
GRID	50	0.98	0.00	0.38
GRID	51	0.97	0.23	0.38
GRID	52	0.96	0.46	0.37
GRID	53	0.93	0.69	0.36
GRID	54	0.89	0.92	0.34
GRID	55	0.69	-0.95	0.21
GRID	56	0.73	-0.71	0.23
GRID	57	0.76	-0.48	0.24
GRID	58	0.78	-0.24	0.24
GRID	59	0.79	0.00	0.24
GRID	60	0.78	0.24	0.24
GRID	61	0.76	0.48	0.24
GRID	62	0.73	0.71	0.23
GRID	63	0.69	0.95	0.21
GRID	64	0.48	-0.97	0.11
GRID	65	0.53	-0.73	0.12
GRID	66	0.56	-0.49	0.13
GRID	67	0.58	-0.24	0.14
GRID	68	0.59	0.00	0.14
GRID	69	0.58	0.24	0.14
GRID	70	0.56	0.49	0.13
GRID	71	0.53	0.73	0.12
GRID	72	0.48	0.97	0.11
GRID	73	0.28	-0.99	0.04
GRID	74	0.33	-0.74	0.05
GRID	75	0.36	-0.49	0.06
GRID	76	0.38	-0.25	0.06
GRID	77	0.39	0.00	0.06
GRID	78	0.38	0.25	0.06
GRID	79	0.36	0.49	0.06
GRID	80	0.33	0.74	0.05
GRID	81	0.28	0.99	0.04
GRID	82	0.07	-1.00	0.01
GRID	83	0.12	-0.75	0.01
GRID	84	0.16	-0.50	0.01
GRID	85	0.18	-0.25	0.01

```
GRID    86        0.19  0.00  0.01
GRID    87        0.18  0.25  0.01
GRID    88        0.16  0.50  0.01
GRID    89        0.12  0.75  0.01
GRID    90        0.07  1.00  0.01
GRID    91        0.00  -0.88 0.00
GRID    92        0.00  -0.66 0.00
GRID    93        0.00  -0.44 0.00
GRID    94        0.00  -0.22 0.00
GRID    95        0.00  0.00  0.00
GRID    96        0.00  0.22  0.00
GRID    97        0.00  0.44  0.00
GRID    98        0.00  0.66  0.00
GRID    99        0.00  0.88  0.00
$**************************************************
$ ELEMENT (Thin Shell Linear Quadrilateral)
$**************************************************
CQUAD4 1    1   1   2   11  10
CQUAD4 2    1   2   3   12  11
CQUAD4 3    1   3   4   13  12
CQUAD4 4    1   4   5   14  13
CQUAD4 5    1   5   6   15  14
CQUAD4 6    1   6   7   16  15
CQUAD4 7    1   7   8   17  16
CQUAD4 8    1   8   9   18  17
CQUAD4 9    1   10  11  20  19
CQUAD4 10   1   11  12  21  20
CQUAD4 11   1   12  13  22  21
CQUAD4 12   1   13  14  23  22
CQUAD4 13   1   14  15  24  23
CQUAD4 14   1   15  16  25  24
CQUAD4 15   1   16  17  26  25
CQUAD4 16   1   17  18  27  26
CQUAD4 17   1   19  20  29  28
CQUAD4 18   1   20  21  30  29
CQUAD4 19   1   21  22  31  30
CQUAD4 20   1   22  23  32  31
CQUAD4 21   1   23  24  33  32
CQUAD4 22   1   24  25  34  33
CQUAD4 23   1   25  26  35  34
CQUAD4 24   1   26  27  36  35
CQUAD4 25   1   28  29  38  37
CQUAD4 26   1   29  30  39  38
CQUAD4 27   1   30  31  40  39
CQUAD4 28   1   31  32  41  40
CQUAD4 29   1   32  33  42  41
CQUAD4 30   1   33  34  43  42
CQUAD4 31   1   34  35  44  43
CQUAD4 32   1   35  36  45  44
CQUAD4 33   1   37  38  47  46
CQUAD4 34   1   38  39  48  47
```

```
CQUAD4 35    1    39    40    49    48
CQUAD4 36    1    40    41    50    49
CQUAD4 37    1    41    42    51    50
CQUAD4 38    1    42    43    52    51
CQUAD4 39    1    43    44    53    52
CQUAD4 40    1    44    45    54    53
CQUAD4 41    1    46    47    56    55
CQUAD4 42    1    47    48    57    56
CQUAD4 43    1    48    49    58    57
CQUAD4 44    1    49    50    59    58
CQUAD4 45    1    50    51    60    59
CQUAD4 46    1    51    52    61    60
CQUAD4 47    1    52    53    62    61
CQUAD4 48    1    53    54    63    62
CQUAD4 49    1    55    56    65    64
CQUAD4 50    1    56    57    66    65
CQUAD4 51    1    57    58    67    66
CQUAD4 52    1    58    59    68    67
CQUAD4 53    1    59    60    69    68
CQUAD4 54    1    60    61    70    69
CQUAD4 55    1    61    62    71    70
CQUAD4 56    1    62    63    72    71
CQUAD4 57    1    64    65    74    73
CQUAD4 58    1    65    66    75    74
CQUAD4 59    1    66    67    76    75
CQUAD4 60    1    67    68    77    76
CQUAD4 61    1    68    69    78    77
CQUAD4 62    1    69    70    79    78
CQUAD4 63    1    70    71    80    79
CQUAD4 64    1    71    72    81    80
CQUAD4 65    1    73    74    83    82
CQUAD4 66    1    74    75    84    83
CQUAD4 67    1    75    76    85    84
CQUAD4 68    1    76    77    86    85
CQUAD4 69    1    77    78    87    86
CQUAD4 70    1    78    79    88    87
CQUAD4 71    1    79    80    89    88
CQUAD4 72    1    80    81    90    89
CQUAD4 73    1    82    83    92    91
CQUAD4 74    1    83    84    93    92
CQUAD4 75    1    84    85    94    93
CQUAD4 76    1    85    86    95    94
CQUAD4 77    1    86    87    96    95
CQUAD4 78    1    87    88    97    96
CQUAD4 79    1    88    89    98    97
CQUAD4 80    1    89    90    99    98
$*****************************************
$     BOUNDARY CONDITION (Restraints)
$*****************************************
SPC1   1      123456 91    92    93    94    95    96    +7
+7     97     98     99
```

```
GRDSET                                    456
$     *     *     *     *     *     *     *     *     *
$****************************************
$        PHYSICAL PROPERTY
$****************************************
PSHELL 1      22     0.08    22      1.00   22     0.83
$****************************************
$        MATERIAL PROPERTY
$****************************************
MAT1   22     2.9E+07        0.30   7.5E-04 6.5E-06
$****************************************
$     MODAL ANALYSIS PARAMETERS
$****************************************
CONVERT MASS    1.
EIGR   1     GIV
ENDDATA
```

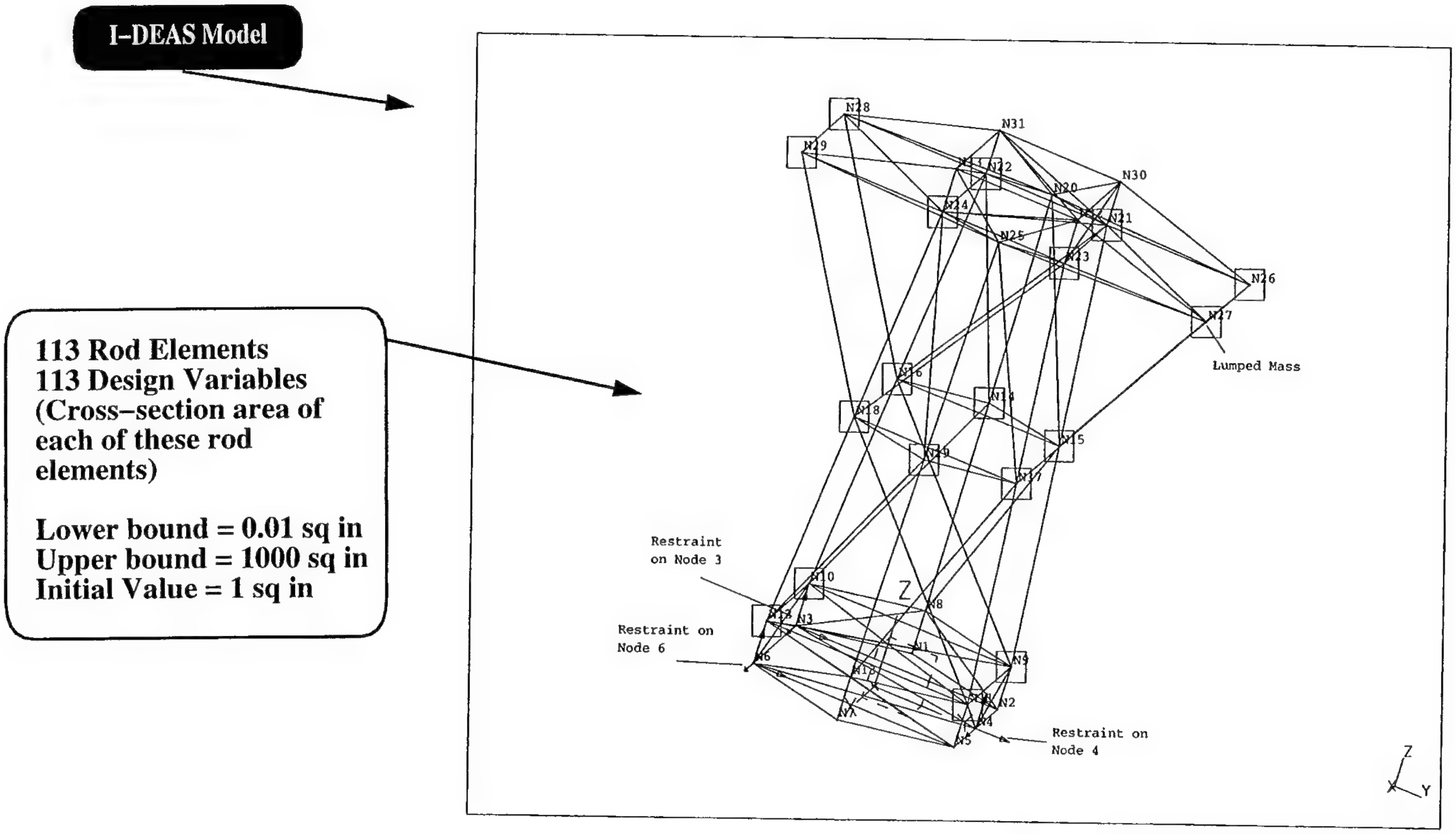


Figure 41. Space Truss Model (ACOSS Structural Model) for Example 10

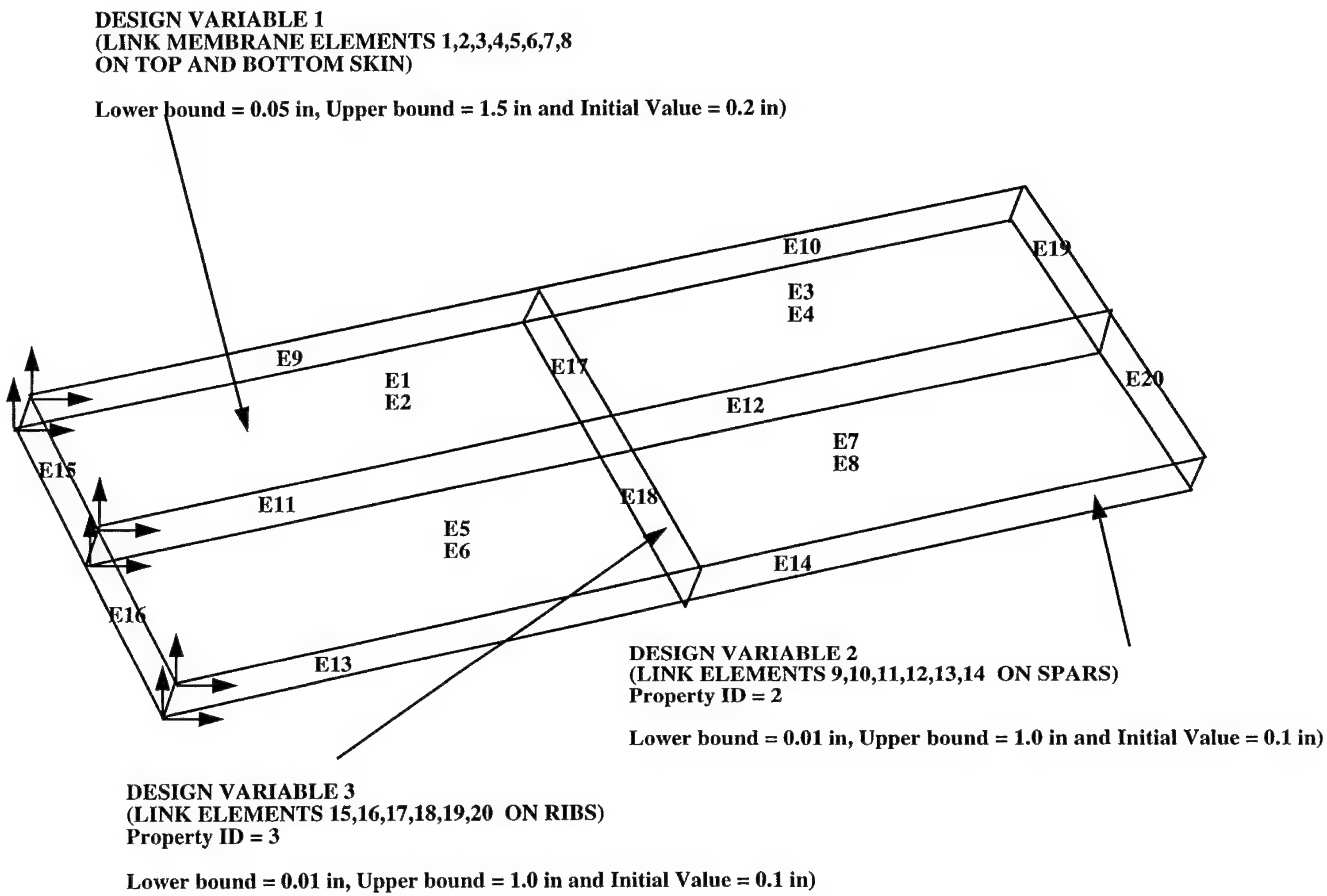


Figure 42. Rectangular Wing Model for Example 10

ASTROS Input for Independent Design Variable (Cross Section Area) Definition (Example 10)

```
$*****************************************************************************
$   DESIGN VARIABLE (CROSS SECTION AREA OF ROD ELEMENT, FOR SPACE TRUSS EXAMPLE IN FIGURE 40)
$*****************************************************************************
DESELM    1      1      CROD  0.01  1000.0  1.0
DESELM    2      2      CROD  0.01  1000.0  1.0
DESELM    3      3      CROD  0.01  1000.0  1.0
DESELM    4      4      CROD  0.01  1000.0  1.0
DESELM    5      5      CROD  0.01  1000.0  1.0
DESELM    6      6      CROD  0.01  1000.0  1.0
DESELM    7      7      CROD  0.01  1000.0  1.0
........................
DESELM    112    112    CROD  0.01  1000.0  1.0
DESELM    113    113    CROD  0.01  1000.0  1.0
```

ASTROS Input for Element Linked Design Variable (Thickness) Definition (Example 10)

```
$*****************************************************************************
$   DESIGN VARIABLE (THICKNESS, FOR RECTANGULAR WING SHOWN IN FIGURE 41)
$*****************************************************************************

DESVARP   1               0.05   1.500   0.2
ELIST     1   CQDMEM1     1      2       3     4     5     6   +1
+1        7   8
DESVARP   2               0.01   1.000   0.1
ELIST     2   CQUAD4      9      10      11    12    13    14
DESVARP   3               0.01   1.000   0.1
ELIST     3   CQUAD4      15     16      17    18    19    20
```

ASTROS Input for Element Linked and Physical Property Linked Design Variable (Thickness) Definition (Example 10)

```
$*****************************************************************************
$   DESIGN VARIABLE (THICKNESS, FOR RECTANGULAR WING SHOWN IN FIGURE 41)
$*****************************************************************************

DESVARP   1               0.05   1.500   0.2
ELIST     1   CQDMEM1     1      2       3     4     5     6   +1
+1        7   8
DESVARP   2               0.01   1.000   0.1
PLIST     2   PSHELL      2
DESVARP   3               0.01   1.000   0.1
PLIST     3   PSHELL      3
```

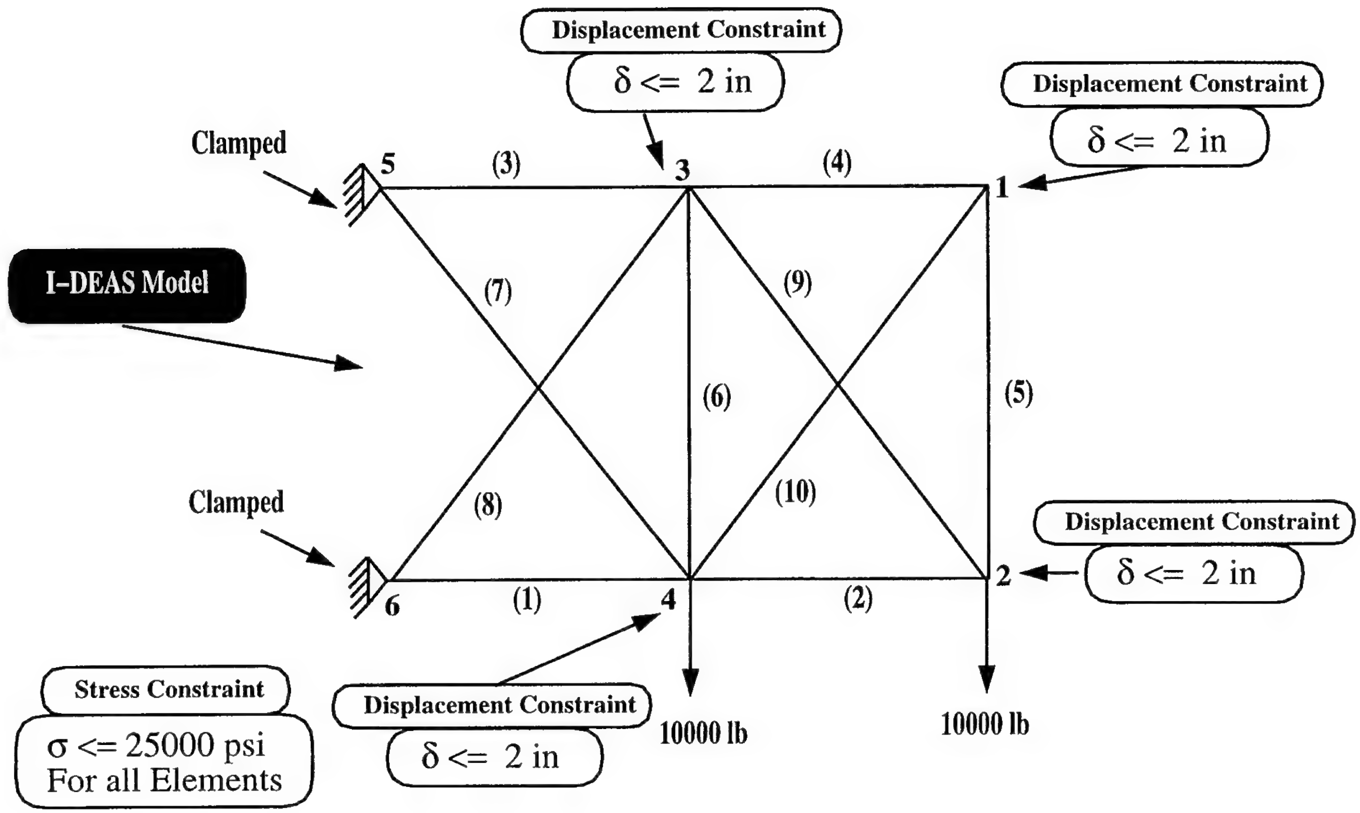


Figure 43. A Ten Bar Truss Model Created in I–DEAS for Example 11

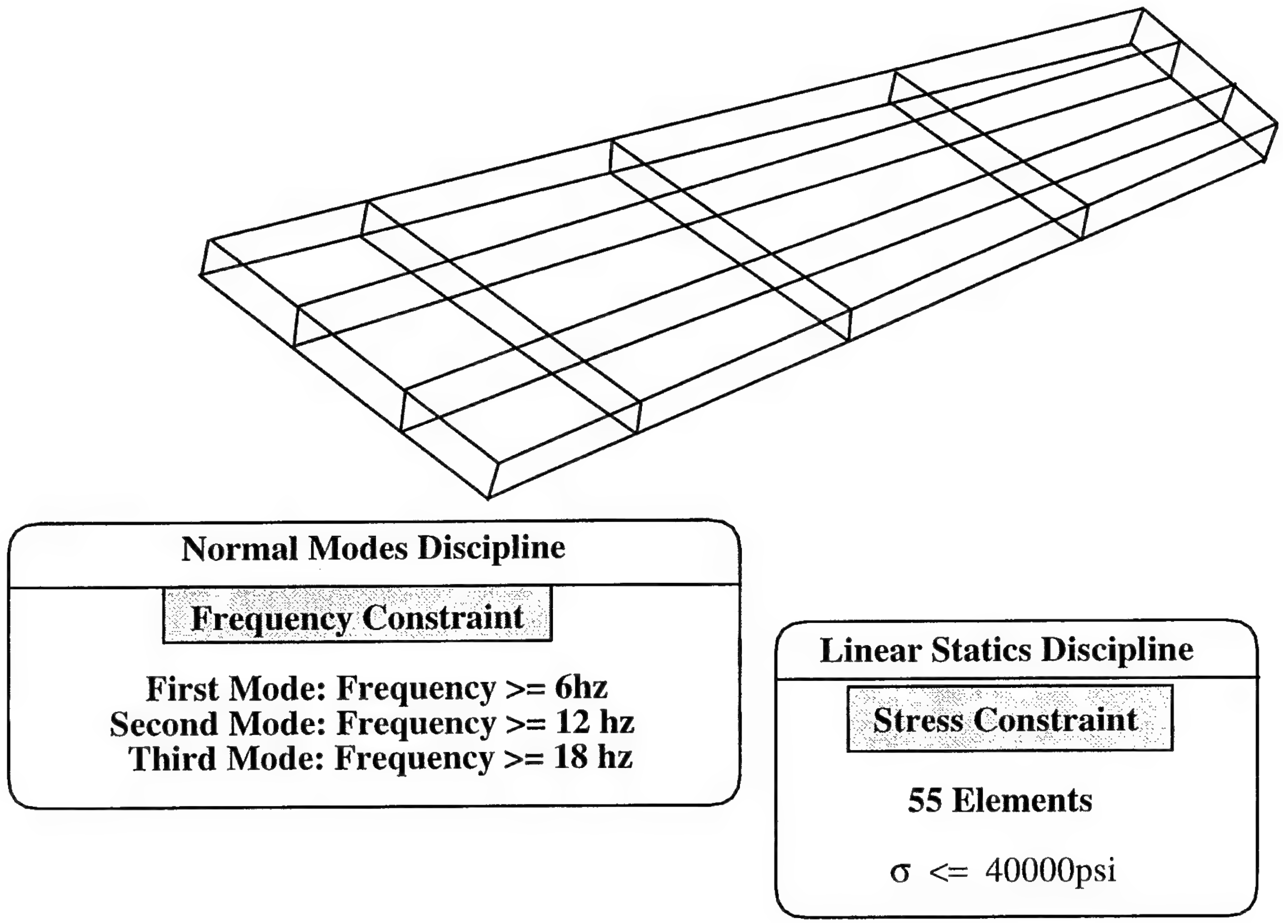


Figure 44. A Wing Model Created in I–DEAS for Example 11

ASTROS Input for Defining Constraint in a Single Discipline (Frequency Constraint for First and Second Mode)

```
$**********************************************************************************
$ DESIGN CONSTRAINT (FREQUENCY CONSTRAINT CONSIDERED FOR SPACE TRUSS STRUCTURE IN FIGURE 40)
$**********************************************************************************
DCONFRQ   1   1   LOWER   2.0
DCONFRQ   1   2   LOWER   3.0
```

ASTROS Input for Defining Multiple Constraints in a Single Discipline (Displacement and Stress Constraints for the Tenbar Truss)

```
$**********************************************************************************
$ DESIGN CONSTRAINT (CONSTRAINTS CONSIDERED FOR TENBAR TRUSS STRUCTURE IN FIGURE 42)
$**********************************************************************************
$   DISPLACEMENT CONSTRAINT ON NODES 1, 2, 3 AND 4
$*********************************************************************************
DCONDSP    100    1    UPPER     2.000    1    2    1.000
DCONDSP    100    2    LOWER    -2.000    1    2    1.000
DCONDSP    100    3    UPPER     2.000    2    2    1.000
DCONDSP    100    4    LOWER    -2.000    2    2    1.000
DCONDSP    100    5    UPPER     2.000    3    2    1.000
DCONDSP    100    6    LOWER    -2.000    3    2    1.000
DCONDSP    100    7    UPPER     2.000    4    2    1.000
DCONDSP    100    8    LOWER    -2.000    4    2    1.000
$*********************************************************************************
$   STRESS CONSTRAINT ON EACH TRUSS ELEMENT (ELEMENT 1 THROUGH 10)
$*********************************************************************************
DCONVM     100    25000.0               ROD              1    THROUGH   +1
+1         10
```

ASTROS Input for Defining Constraints in a Multiple Discipline (Stress and Natural Frequency Constraints for the Wing Model)

```
$*********************************************************************************
$   DESIGN CONSTRAINT (STRESS AND FREQUENCY CONSTRAINT CONSIDERED FOR WING STRUCTURE)
$*********************************************************************************
$   STRESS CONSTRAINT ON EACH ELEMENT (ELEMENT 1 THROUGH 55)
$*********************************************************************************
DCONVM     100    40000.0               QUAD4            1    THROUGH   +1
+1         55
$**********************************************************************************
$ DESIGN CONSTRAINT (FREQUENCY CONSTRAINT CONSIDERED FOR ALL THREE MODES)
$**********************************************************************************
DCONFRQ    200    1    LOWER    6.0
DCONFRQ    200    2    LOWER    12.0
DCONFRQ    200    3    LOWER    18.0
```

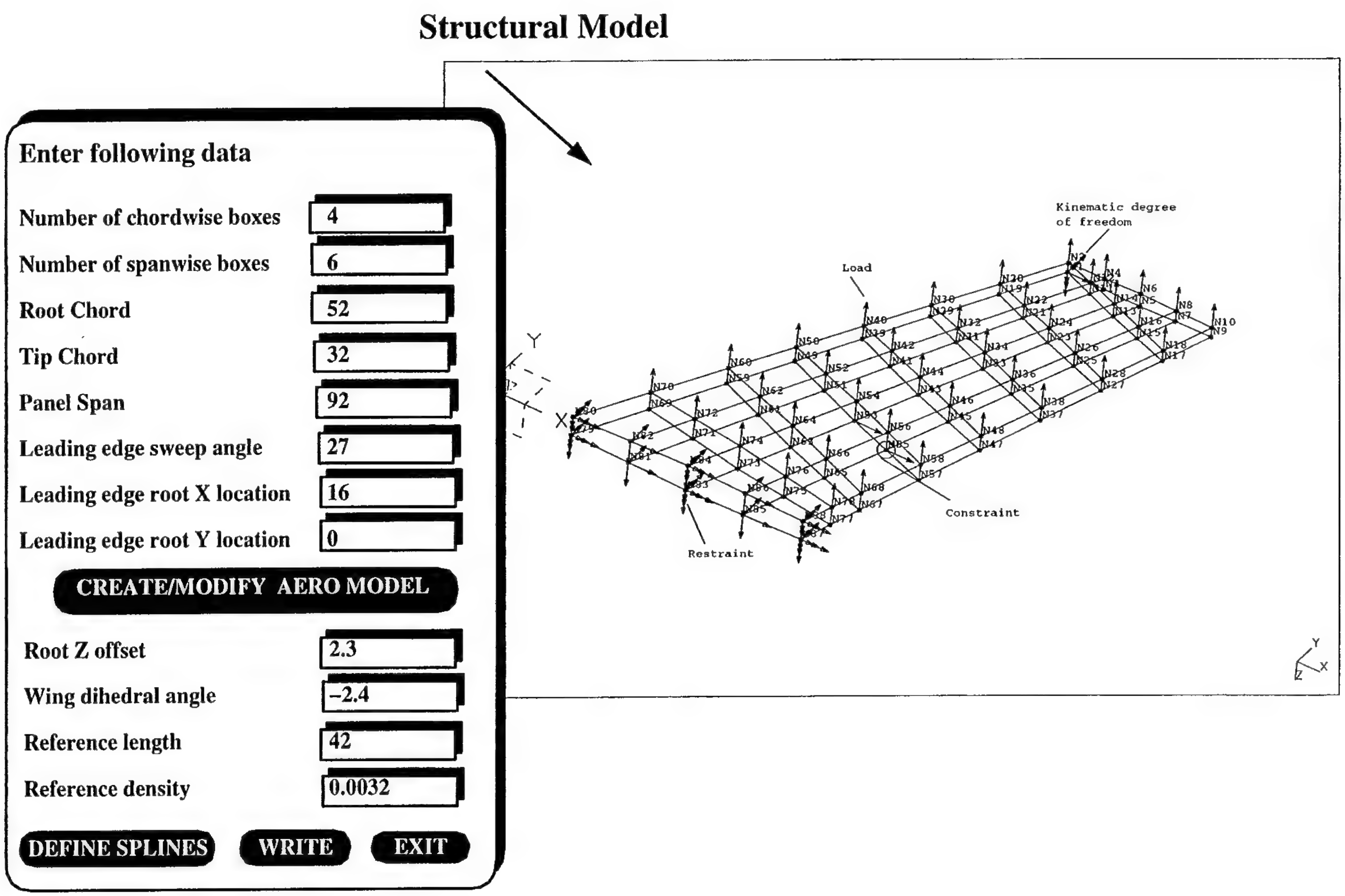


Figure 45. Aero Model Generation for an Intermediate Complexity Wing (Example 12)

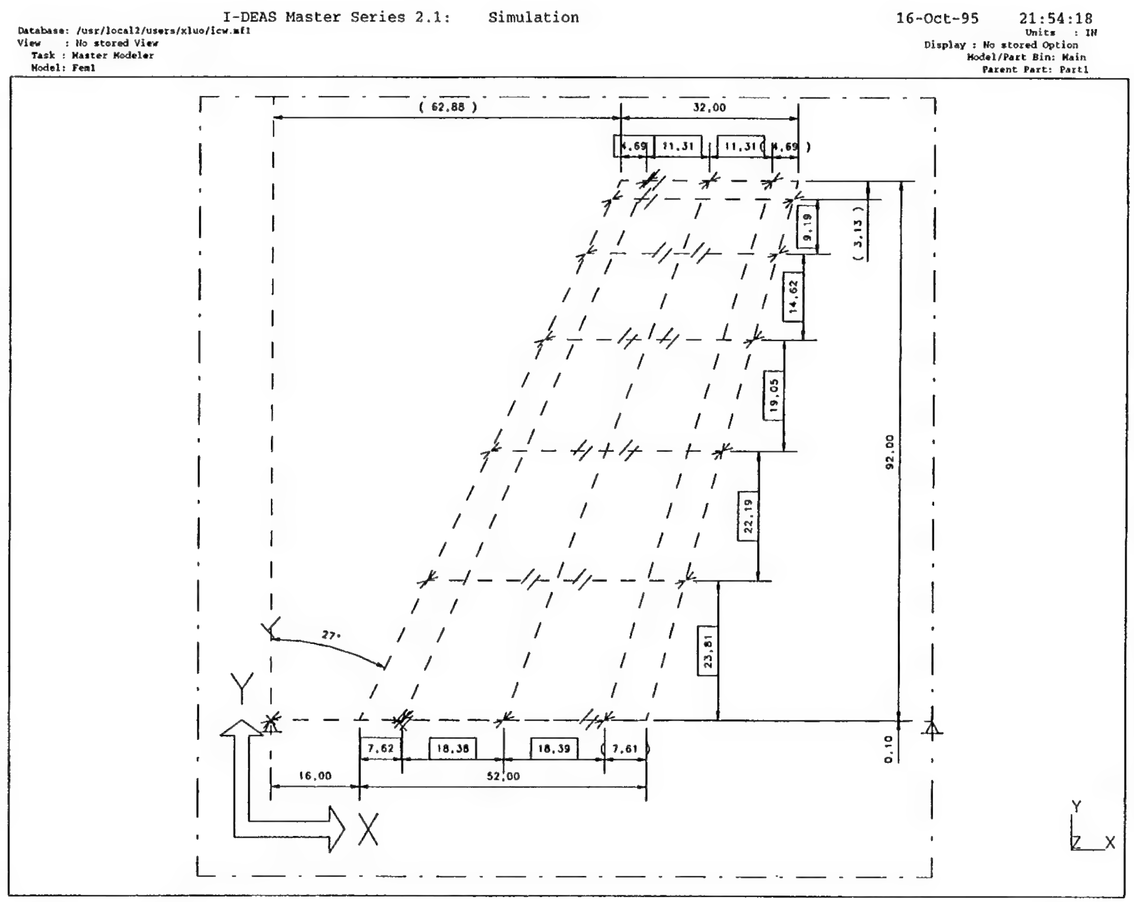


Figure 46. Aero Model Generated by MIDAS for an Intermediate Complexity Wing (Example 12)

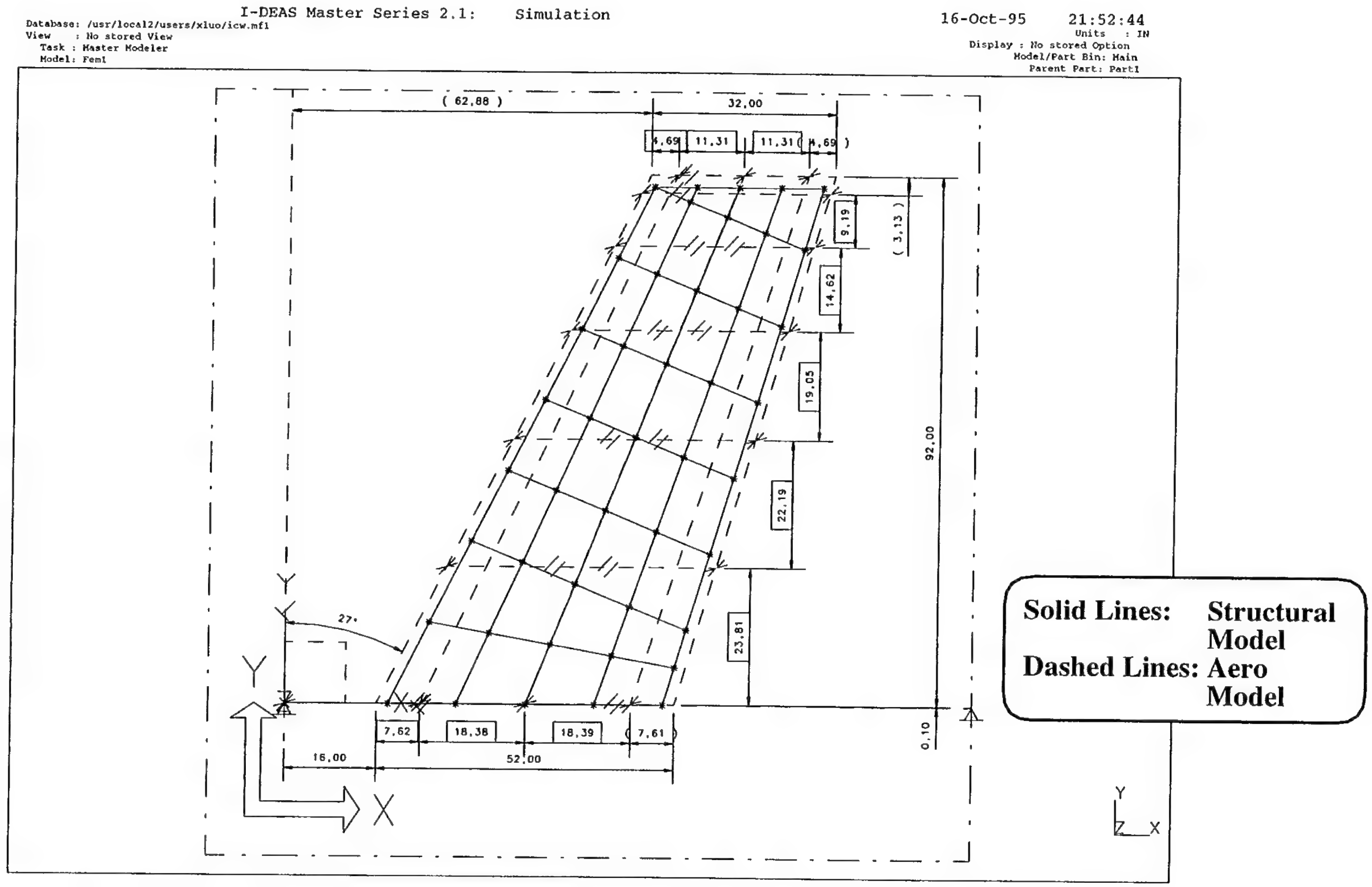


Figure 47. Aero Model Overlayed on the Structural Model of the Intermediate Complexity Wing

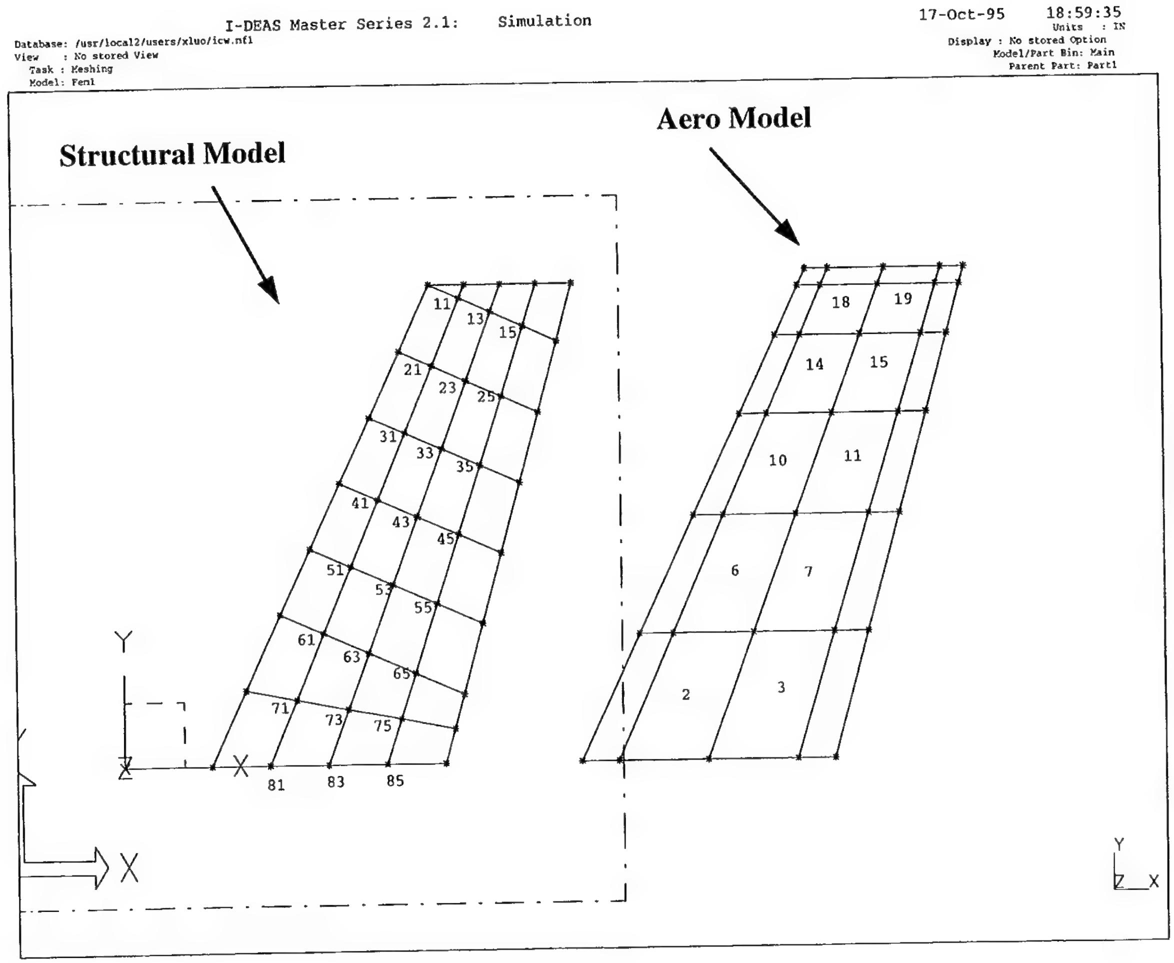


Figure 48. Aero Model Displayed Along with the Structural Model for Defining the Spline

ASTROS Input Aero Model (Example 12)

```
$***************************************************************************
$  AERO MODEL
$***************************************************************************
$  AERODYNAMIC PHYSICAL DATA
$***************************************************************************
AERO            42.0          0.0032
$***************************************************************************
$  AERODYNAMIC MACROELEMENT
$***************************************************************************
CAERO1   1001                                              101     102    +CONTI
+CONT1   16.000   0.100   2.500   52.000   62.876   92.100   6.195   32.000
$***************************************************************************
$  SPANWISE AND CHORDWISE DIVISIONS
$***************************************************************************
AEFECT   101   0.0    0.146    0.500    0.854    1.0
AEFECT   102   0.0    0.259    0.500    0.707    0.866    0.966    1.0
$***************************************************************************
$  SPLINE DEFINITION
$***************************************************************************
SPLINE1    1001     1001     2      19      201     0.0
SET1       201   11   13   15   21   23   25   +CONTI
+CONTI           31   33   35   41   43   45   +CONTI
+CONTI           51   53   55   61   63   65   +CONTI
+CONTI           71   73   75   81   83   85
```

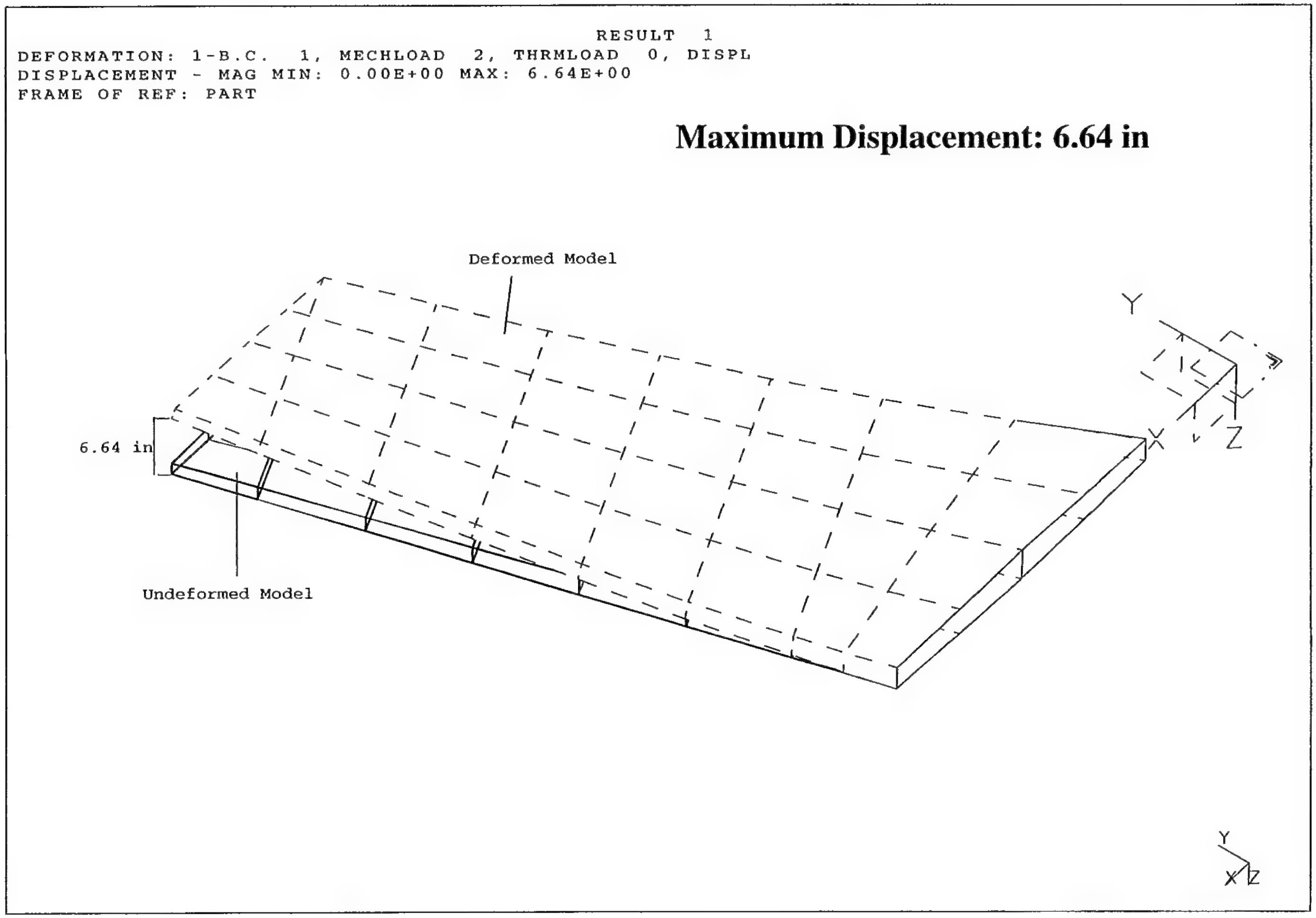
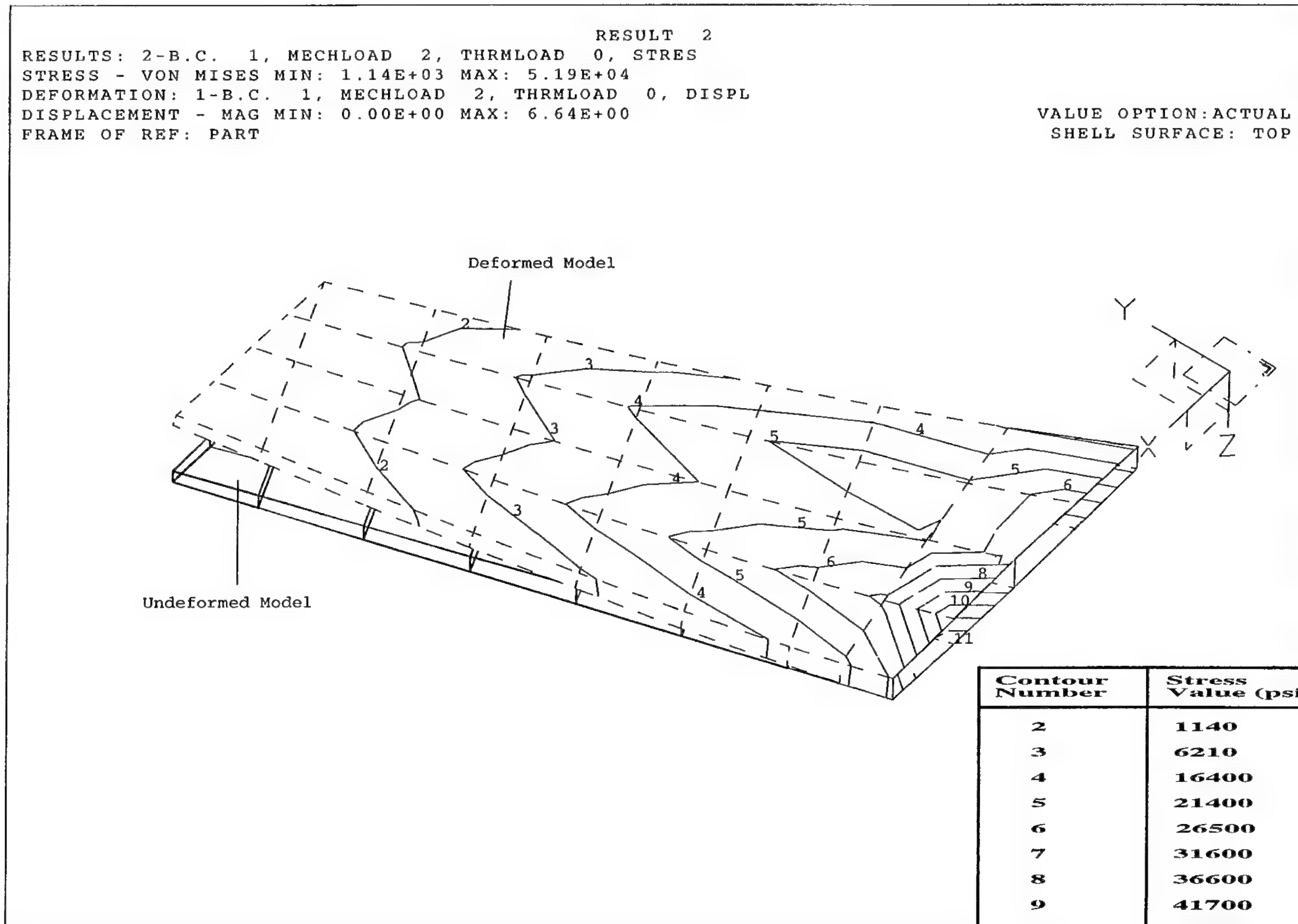
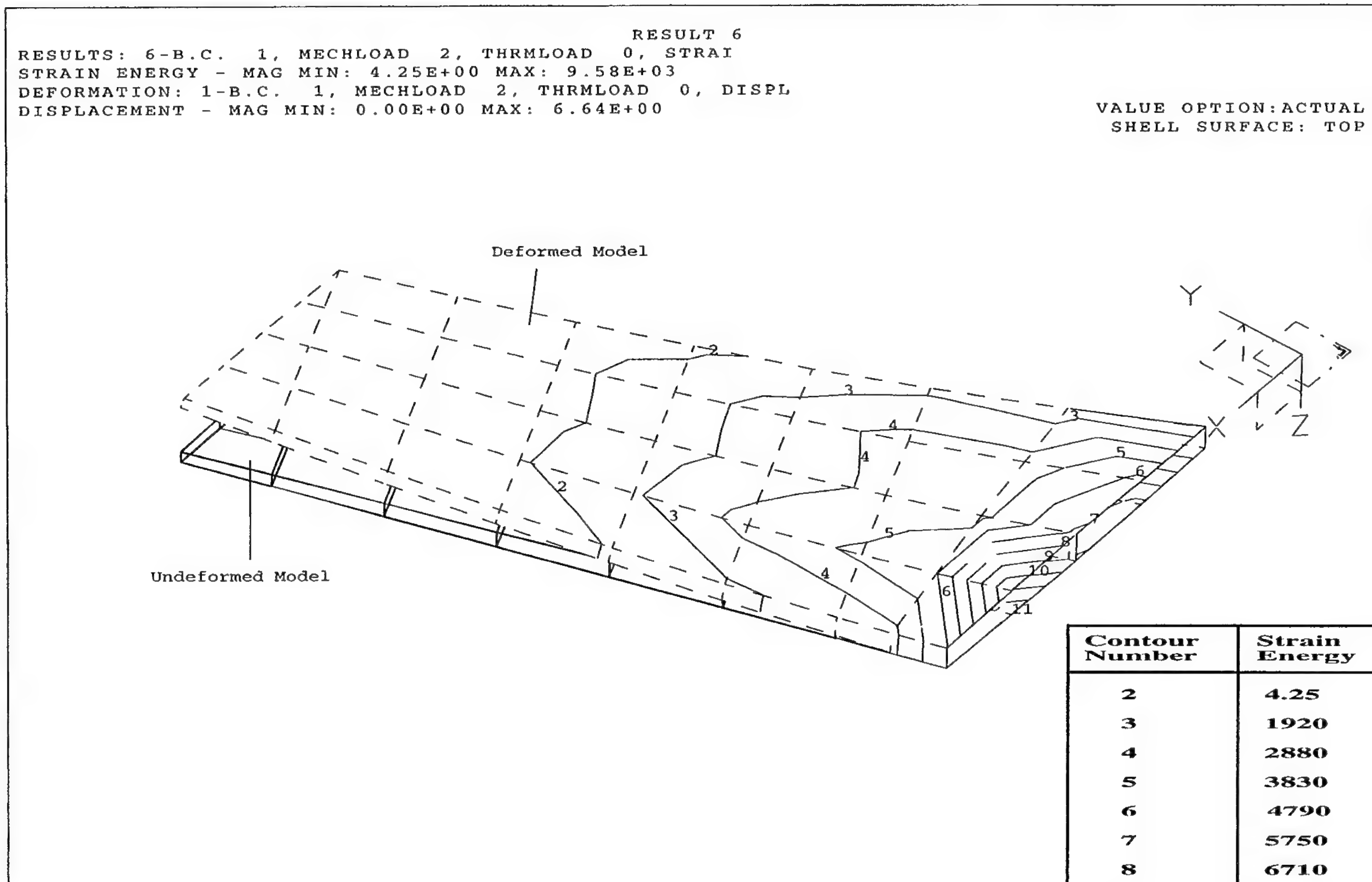


Figure 49. Nodal Displacements Extracted by MIDAS for ASTROS



Contour Number	Stress Value (psi)
2	1140
3	6210
4	16400
5	21400
6	26500
7	31600
8	36600
9	41700
10	46800
11	51900

Figure 50. Element Stresses (Von Mises) Extracted by MIDAS for ASTROS



Contour Number	Strain Energy
2	4.25
3	1920
4	2880
5	3830
6	4790
7	5750
8	6710
9	7660
10	8620
11	9580

Figure 51. Strain Energy Extracted by MIDAS for ASTROS

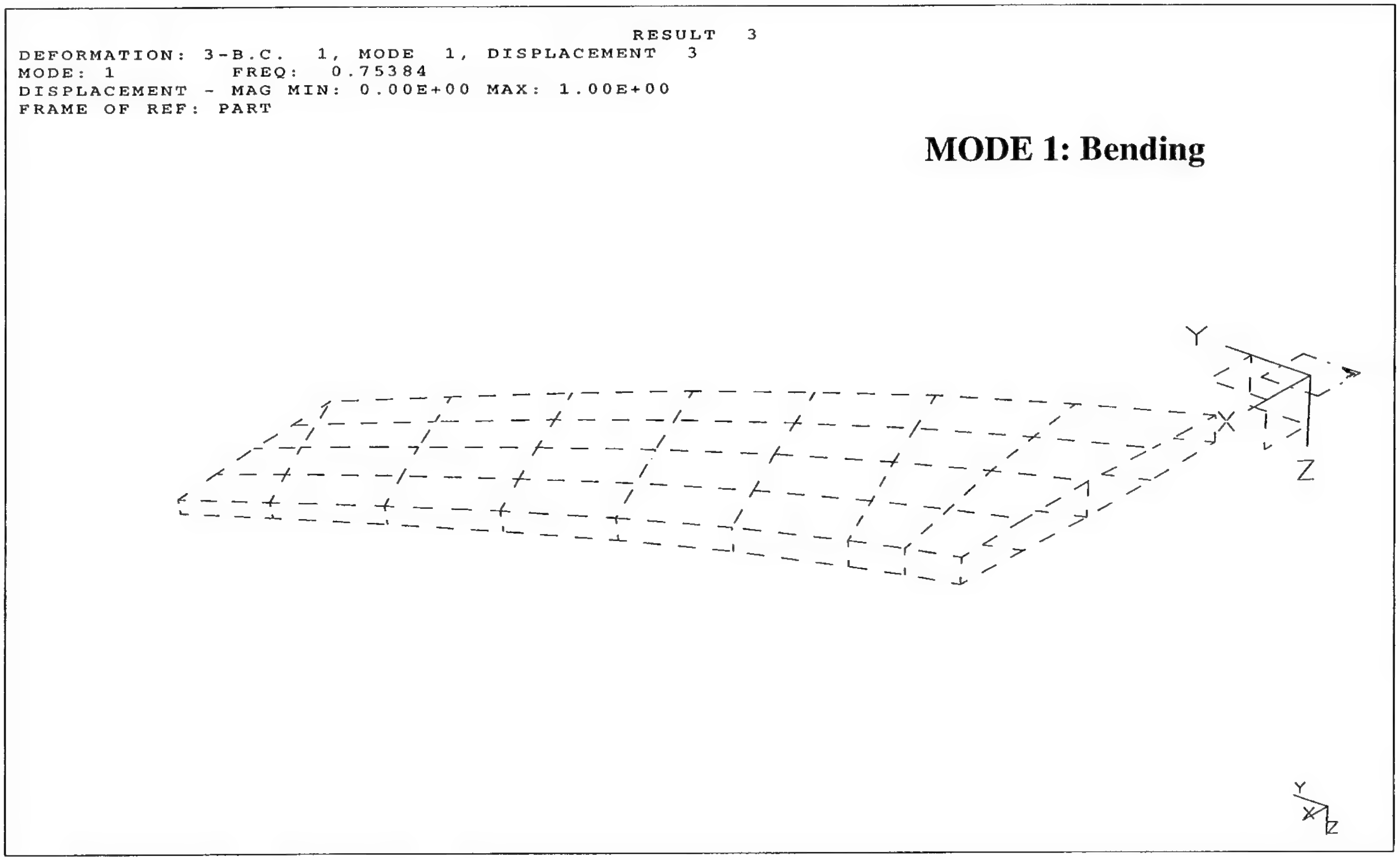


Figure 52. Mode Shape (For Mode 1, Bending)

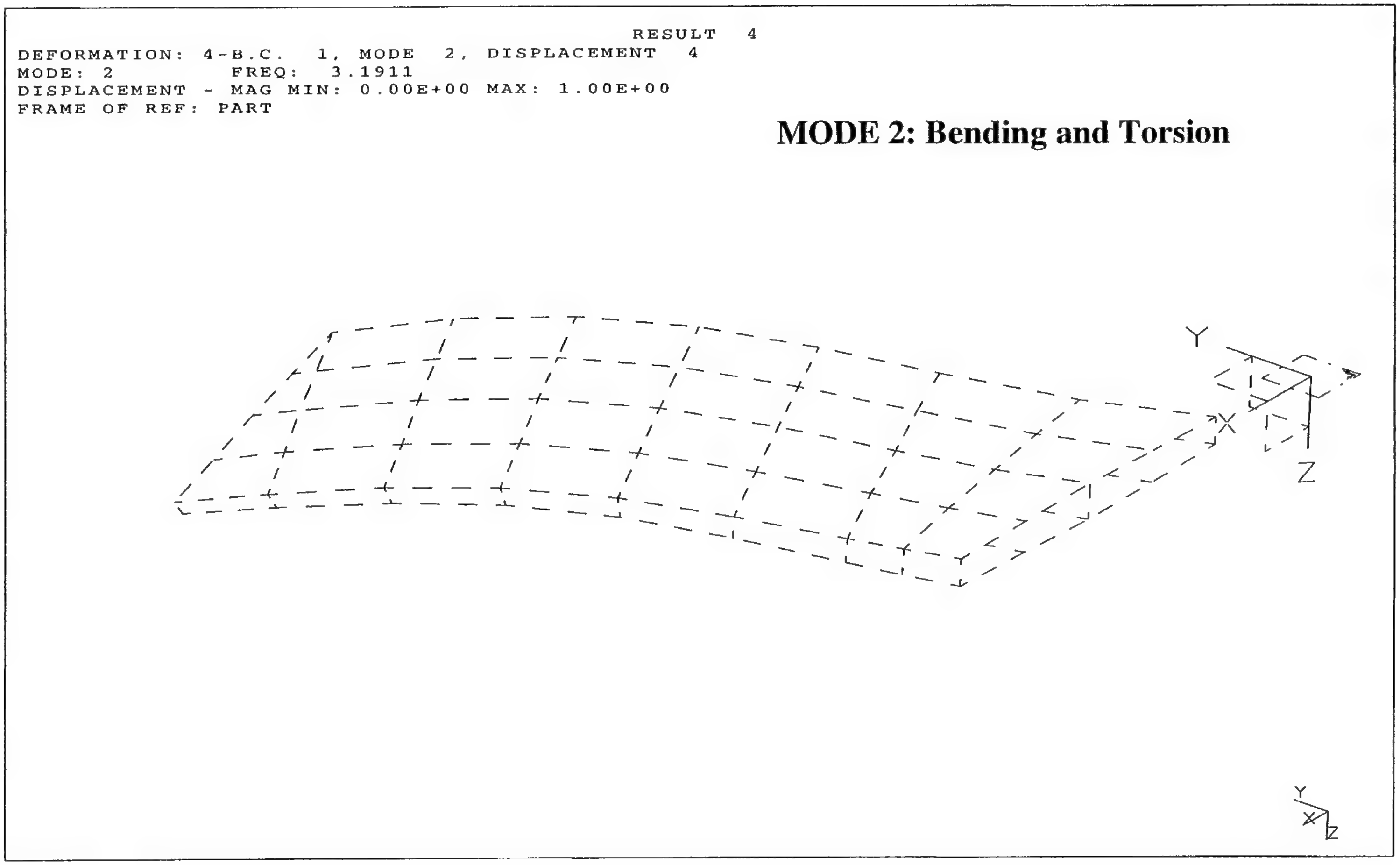


Figure 53. Mode Shape (For Mode 2, Bending and Torsion)

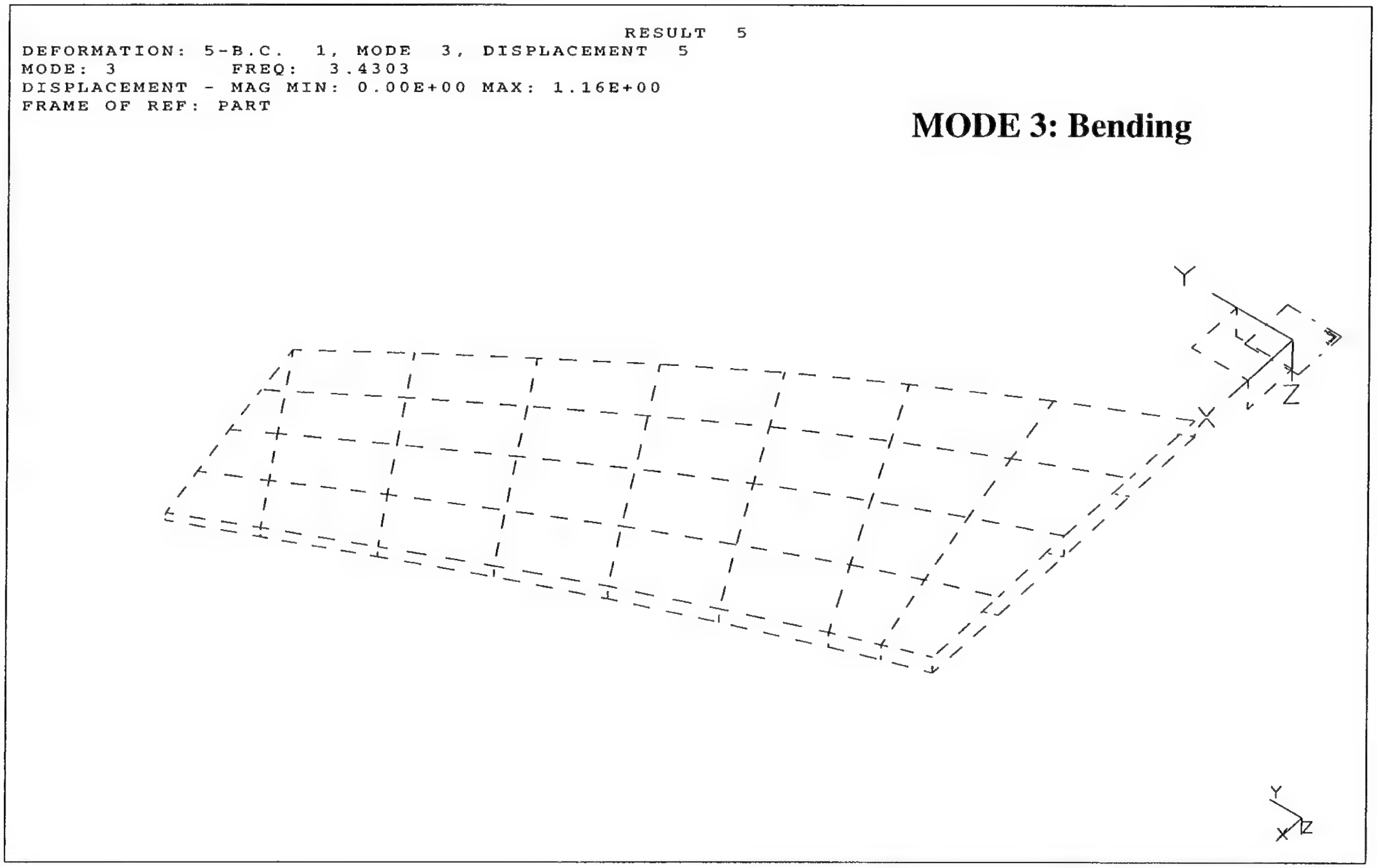


Figure 54. Mode Shape (For Mode 3, Bending)

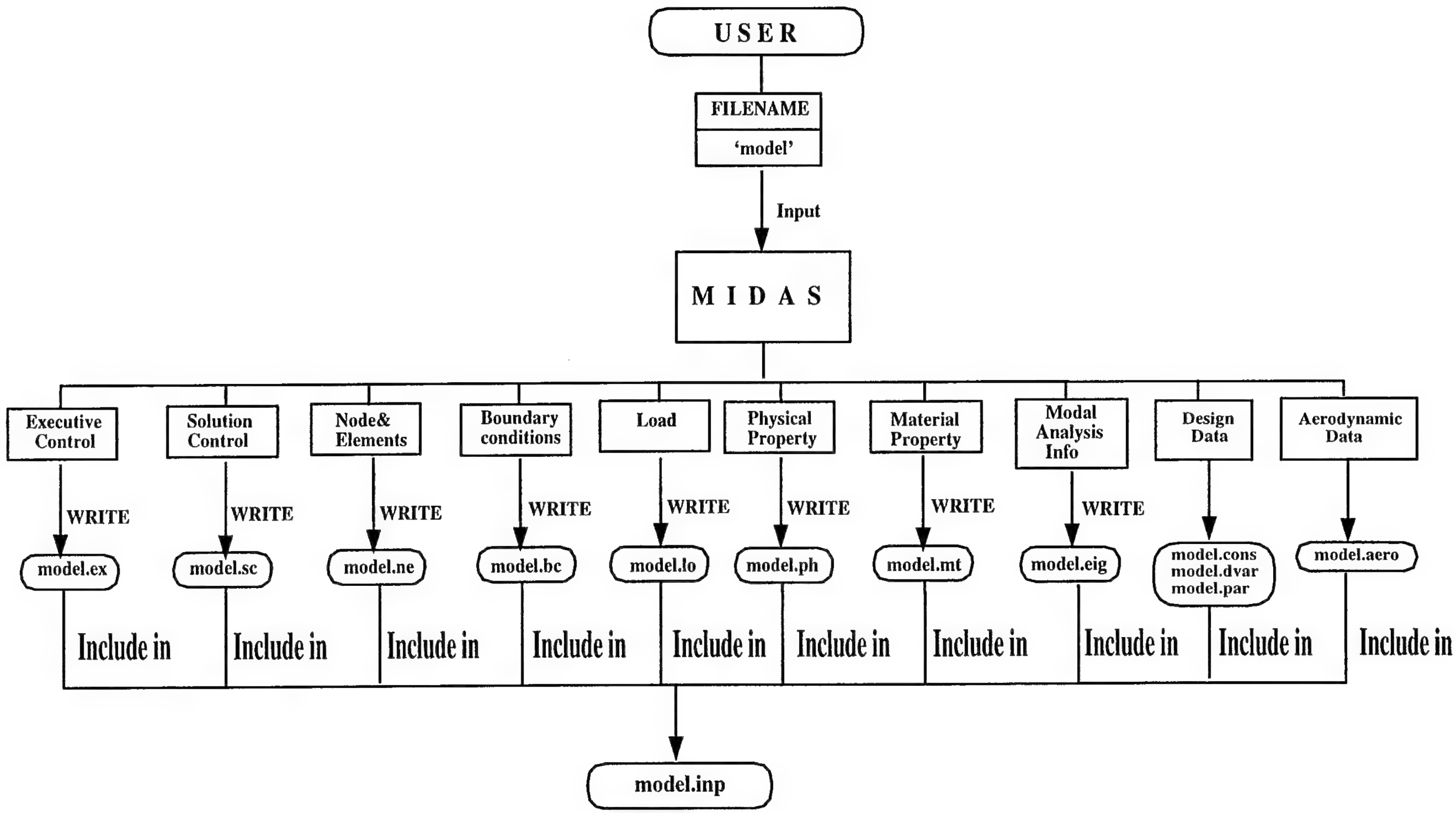


Figure 55. File Management in MIDAS

Chapter Five

Summary Remarks

MIDAS was developed keeping the user's greatest needs in mind. The special features of ASTROS which can not be accommodated in commercially available graphics packages were addressed as priority items. Emphasis was placed on displaying the aerodynamic and structural models simultaneously (superimposed) and also on generation of design optimization cards. Significant effort was directed towards the graphic display of most commonly used static and normal mode analysis results. MIDAS was enabled to handle multiple disciplines, multiple boundary conditions and multiple load cases, which are typical of an ASTROS input stream. The system is currently under development for additional features such as display of design optimization iterations, optimal variable distributions and flutter modes.

MIDAS was developed to integrate ASTROS with I-DEAS Master Series using its Open Architecture tools. It acts as a very powerful graphic pre- and post-processor and has simplified modeling procedures for ASTROS. Using MIDAS, a designer can build the model in fraction of the time compared with what it usually takes if traditional text-oriented methods are followed. It has added visualization features, which has been lacking in ASTROS until now. Also the possibility of modeling errors have been greatly minimized due to the error checking capabilities of I-DEAS and error checking routines in MIDAS. Most importantly, it has provided an integrated platform for conducting design studies and has eliminated the need on the part of the user to go various platforms to carry out pre- and post-processing. Also with an integrated system time is saved in model transportation between different software.

References

1 Benzley, S. E., Merkley, K., Blacker, T, D., and Schoof, L., "Pre- and Post-processing for the Finite Element Method", Finite Elements in Analysis and Design, Vol. 19, No. 4, 1995, 243-260

2 Tworzyollo, W. W. and Oden, J. T., "Towards an Automated Environment in Computational Mechanics", Computer Methods in Applied Mechanics and Engineering, Vol. 104, No. 1, 1993, 87-143

3 Tarrago, J. A., Canales, J., and Arias, A., " CODYSIS: An Integrated System for Optimal Structural Design", Computers and Structures, Vol. 52, No. 6, 1994, 1221-1241

4 Marlowe, J. M. and Lindell, M. C., "NASSTAR: An Instructional Link Between MSC / NASTRAN and STAR", Proceedings of the 11th International Modal Analysis Conference, Kissimmee, FL, Feb 1-4, 1993, 1471-1477

5 Jiazhen, H., "INTOAD - An Interface Between IRM and ADINA", Computers and Structures, Vol. 47, No. 4-5, 1993, 751-756.

6 Rasdorf, W. J., Spainhour, L. K., Patton, E. M. and Burns, B. P., "A Design Environment for Laminated Fiber-Reinforced Thick Composite Materials", Engineering with Computers, Vol. 9, No. 1, 1993, 36-48

7 Arabshahi, S., Barton, D. C. and Shaw, N. K., "Steps towards CAD-FEA Integration", Engineering with Computers, Vol. 9, No. 1, 1993, 17-26

8 Stone, T. T., Santos, J. L. T. and Haug, E. J., " Interactive Pre-processor For Structural Design Sensitivity Analysis and Optimization", Computers and Structures, Vol. 34, No. 3, 1990, 375-385

9 Miki, M and Murotsu, Y., "Object-Oriented Approach to Modeling and Analysis of Truss Structures", AIAA Journal, Vo. 33, 1995, 348-354

10 Caradona, A, Klapka, I. and Geradin, M., "Design of a New Finite Element Programming Environment", Engineering with Computers, Vol. 11, No. 4, 1994, 365-381

11 Pidaparti, R. M. V. and Hudli, A. V., "Dynamic Analysis of Structures Using Object-Oriented Approach", Computers and Structures, Vol. 49, No. 1, 1993, 149-156

12 Yu, G. and Adeli, H., "Object-Oriented Finite Element Analysis Using EER Model", Journal of Structural Engineering, Vol. 119, No. 9, 1993, 2763-2781

13 Chandu, S. V. L., Rajagopalan, H. and Grandhi, R, V., "VAASEL Expert System", Final Report, Center for Artificial Intelligence Applications, Dayton, OH, 1993

14 Labrie, R., Thilloy, P. A. and Moll, G. H., "Expert Assistant to Monitor Finite Element Simulations", Mathematics and Computers in Simulation, Vol. 36, No. 4-6, 1994, 413-422

15 Kang, E. and Haghighi, K., "Intelligent Finite Element Mesh Generation", Engineering with Computers, Vol. 11, No. 2, 1995, 70-82

16 Wang, Q., Rao, M. amd Zhou, J., "Intelligent Systems Approach to Conceptual Design", International Journal of Intelligent Systems, Vol. 10, No. 3, 1995, 259-294

17 Chey, G. and Zeng, Y., "Strategies for Automatic Finite Element Modeling", Computers and Structures, Vol. 44, No. 4, 1992, 905-909

18 Zhang, D., Cheng, G. and Gu, Y., "Automatic Generation of Quadrilateral Mapping Elements and Applications of Shape Optimization Software", Computers and Structures, Vol. 45, No. 4, 1992, 697-705

19 Grandhi, R. V., Sarma, H. K., and Hankey, W. K., "ETOP: Expert System for Trajectory Optimization", International Journal of Systems Automation: Research and Applications (SARA), 2, 1991, pp. 63-81.

20 Daniel, W. J. T., "An Expert System for Validation of Displacement and Rotation Boundary Conditions on Finite Element Models", Computers and Structures, Vol. 39, No. 1/2, 1991, 105-119

21 Swanson, J, A., "Keynote Paper: Current and Future Trends in Finite Element Analysis", Intl. J Comput. Appl. Technol, Vol. 73, No. 6, 1994, 108-117

22 George, J., Peterson, J. and Sorthard, S., "Multidisciplinary Integrated Design Assis-

tant for Spacecraft (MIDAS)", 36th AIAA / ASME / ASCE / AHS / ASC Structural, Structural Dynamics and Materials Conference, New Orleans, LA, 1995, 1790-1799

23 Chandu, S. V. L., Rajagopalan, H. and Grandhi, R, V., "Reliability Based Structural Analysis and Optimization in X Window Environment", 36th AIAA/ ASME/ ASCE/ AHS/ ASC Structural, Structural Dynamics and Materials Conference, New Orleans, LA, 1995, 1364-1370

24 Zheng, Y., Lewis, R. W. and Gethin, D. T., "FEView: An Interactive Visualization Tool for Finite Elements", Finite Elements in Analysis and Design, Vol. 19, No. 4, 1995, 261-294

25 Fox, E., "The Pratt & Whitney Probabilistic Design System", 35th AIAA/ ASME/ ASCE/ AHS/ASC SDM Conference, Hilton Head, SC, Part 2, 1994, 1075-1085.

26 Karupurapu, R. and Bathurst, R. J., "Development of a Finite Element Analysis Post-processing Program", Adv Eng Software, Vol. 16, No. 1, 1993, 15-22

27 Kumar, R., Aggarwal, J. and Sharma, S. P., "Interactive Graphic Software for Analysis of Rotationally Symmetric Structures", Computers and Structures, Vol. 46, No. 6, 1993, 1119-1123

28 Parkinson, A. R., Balling, R. J. and Free, J. C., "OPTDESX: An X Window Based Optimal Design Software System", Fourth AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization, Cleveland, OH, 1992, 559-566

29 Lucas, S. and Davis, R., "MacPASCO: A Macintosh Based Interactive Graphic Pre-processor for Structural Analysis and Sizing", 32nd AIAA/ ASME / ASCE / AHS / ASC Structural, Structural Dynamics and Materials Conference, Baltimore, MD, 1991, 612-626

30 Santose, J, L, T., Godse, M. M., Chang, K. H., "Interactive Post-processor for Structural Design Sensitivity Analysis and Optimization, Sensitivity Display and What-If Study", Computers and Structures, Vol. 35, No. 1, 1990, 1-13

31 Berlage, T., "OSF/Motif as a User Interface Standard", Comput. Stand. Interfaces,

Vol. 17, No. 1, 1995, 99-106

32 Yurkovich, R., "The Use of Taguchi Techniques with the ASTROS Code for Optimum Wing Structural Design", 35th AIAA/ ASME/ ASCE/ AHS/ASC SDM Conference, Hilton Head, SC, Part 2, 1994, 1334-1342.

33 Stevens, D. C. "An Interactive Model Generator for Structural Optimization with Aeroelastic Constraints", University of Oklahoma, Norman, OK, 1993.

34 Luo, X. and Grandhi, R. V., "ASTROS for Reliability Based Multidisciplinary Structural Analysis and Optimization", 36th AIAA/ASME/ASCE/AHS/ASC Structural, Structural Dynamics and Materials Conference, New Orleans, LA, 1995, 93-102

35 Sarma, H. K., Grandhi, R. V., and Taylor, R. F. "Expert System for Multidisciplinary Analysis and Optimization Using ASTROS", Journal of Artificial Intelligence for Engineering Design, Analysis and Manufacture, Vol. 4 No. 2, 1990, 117-128

36 Neill, D. J. and Herendeen, D. L., ASTROS Users Manual, WL - TR - 93 - 3025, Wright Patterson Air Force Base, Dayton, OH, 1993

37 Johnson, E. H. and Neill, D. J., ASTROS Applications Manual, AFWAL - TR - 88 - 3028, Wright Patterson Air Force Base, Dayton, OH, 1988

38 Johnson, E. H. and Venkayya, V. B., ASTROS Theoretical Manual, AFWAL - TR - 88 - 3028, Wright Patterson Air Force Base, Dayton, OH, 1988

39 Neill, D. J., Herendeen, D. L. and Hoesly, R. L., ASTROS Programmer's Manual, WL - TR - 93 - 3038, Wright Patterson Air Force Base, Dayton, OH, 1988

40 I-DEAS Master Series, Release 2, Exploring I-DEAS Simulation, Structural Dyanmics Research Corporation (SDRC), Milford, OH, 1994

41 I-DEAS Master Series Open Architecture, Training Coruse, Structural Dyanmics Research Corporation (SDRC), Milford, OH, 1994

42 I-DEAS Master Series Open Architecture, Reference Manual, Structural Dyanmics Research Corporation (SDRC), Milford, OH, 1994